DRAGON USER

The independent Dragon magazine

September 1987

Contents

Editorial

WELL, I was wrong about Wales. It rained in the morning and in the afternoon. And all night as well. And every day little troops of walkers and cyclists dripped into our village in search of a pint of anything but water. Such is the indomitable spirit that Wales inspires.

It's early in the month here and the Special Offer books are selling fast, but because I am an optimist (or is it pessimist?) I am running the coupons again so that anyone who wants extra copies, to pass on the page to a friend, or didn't want to hack up last month's classified page gets another chance. If all goes well we may be able to arrange something else along the same lines.

I see Paul Grade has been laying into users who tape copies of NDUG software. Quite right too. Duplicating other people's produce to save a couple of quid here and there is bad news. Ripping off anyone's software is bad news these days. You might say 'well, I wouldn't buy it anyway', but it sets a bad example, and Dragon suppliers, who are playing to a small market, need your support.

Think twice, take my advice. You could be cutting your own network from under you.

Telephone number
(All departments)
437-4343

Editor
HELEN ARMSTRONG

Production Editors
DAVID PRIVETT/LINDA SLOMAN

Editorial Secretary
CAROL FRITH

Advertisement Manager
ATHENA PEERMAN

Administration
ANNE MARIE ALLEN

Marketing Manager
HELEN PERRY

Managing Editor
PETER WORLOCK

Publishing Director
JENNY IRELAND

Subscriptions
UK £14 for 12 issues
Overseas (surface) £20 for 12 issues
ISSN 0265-177. Telex: 296275
Dragon User, 12/13 Little Newport Street, London WC2H 7PP
US address: c/o Business Press International, 205 East 42nd St, New York, NY 10017
Published by Scot Press Ltd.

Typesetting and Production by Artext Limited, London NW1.
Printed by Headley Brothers Ltd. Ashford, Kent
Registered at the Post Office as a newspaper.

How to submit articles

The quality of the material we can publish in *Dragon User* each month will, to a very great extent depend on the quality of the discoveries that you can make with your Dragon. The Dragon computer was launched on to the market with a powerful version of Basic, but with very poor documentation.

Articles which are submitted to *Dragon User* for publication should not be more than 3000 words long. All submissions should be typed. Please leave wide margins and a double space between each line. Programs should, whenever possible, be computer printed on plain white paper and be accompanied by a tape of the program.

We cannot guarantee to return every submitted article or program, so please keep a copy. If you want to have your program returned you must include a stamped addressed envelope.

Letters

Adventure extra

I would like to wish Peter Gerrard good luck in his takeover of the Adventure column and hope that I can become as good a friend to him as I was to his brother Mike. May his light never go out.

A little tip for those stuck in *Tanglewood*: drink in western well to use the springboard.

Simon 'The Solver' Hargrave
Crawley Hill Farm
Uley
Dursley
Glos. GL11 5BH

At the last count Mike's light was still glowing gently over his ex-column, because the signwriter hasn't been round to fix it. But we're having it sent along to *The Guardian* as soon as possible.

Machine code please

EVERY month I enjoy reading your brilliant magazine, but I think that something is still to come.

The one thing I'm looking for every month is machine code for the more advanced user. I think and know that all we Dragon users have had the computer for a while and know something about machine code, and I don't think we are interested in Basic for the beginner.

When I say machine code, I mean machine code that includes machine code in conjunction with the disc too (FX. How to load and save from MC). I must say that even the Basic programs printed over the years are of a very high quality.

I would also like to read more about hardware projects. For example, I'm right now building a battery-backup-real-time-clock and a RS232 interface into my Dragon 32. I have built a PIA which I use very much when I communicate with other computers. I would like to tell you more about the projects if you would like it.

Jakob Hoffman
2 Ellekonebakken
8800 Viborg
Denmark

We are not certain quite when, but another series on machine code is planned in the future.

We would actually like to run more hardware projects, but we can only in fairness consider small projects, as the potential for getting into trouble on an ambitious modification is almost limitless; this kind of thing is best left to specialist hardware magazines. Anybody who wants to send us a hardware project for consideration: please send a detailed description of what the device does, a COMPLETE set of drawings, including any PCB layout involved, and a complete list of parts used, along with the names and addresses of current suppliers, and cost.

Then we can think about publishing!

Every month we will be shelling out a game or two, courtesy of Microdeal, to the reader/s who send the most interesting or entertaining letters. So send us your hints and your opinions, send us your hi-scores and suggestions. Send us your best Dragon stories. What d'you think we are, mind readers?!

EXTRA PUFF

Music for the Millions

SOME time ago I wrote a letter of thanks to all the Dragon users who responded to a request for help with 'Puff', our tame Dragon at Westgarth, a short-stay home for children with a mental handicap. The response was overwhelming, and several people have kept in touch with us and provide continuing help and advice. I acquired a Dragon 64 for my own use in creating programs for the children, along with Microdeal's *Composer* utility which, as an ex-professional musician, I found invaluable for producing 'proper' music on the Dragon. It does, however require some musical knowledge to produce the best results, particularly when moving graphics note by note. Unless the note are all the same length, a very jerky effect can result, as well as making the tune sound rather odd with the pauses as the graphics are moved!

By way of a 'thank you' to Dragon users, I am developing a library of *Composer* tunes, both Basic DATA statements and M/C files so that programs with little musical knowledge can have access to a ready mode source of variety of music, from short 'jingles' and record tunes compatible with moving graphics, to full-length classical pieces just for listening to, nursery rhymes, Christmas carols and TV themes, etc. The Basic DATA lines are commented with REMs to enable lists of tunes to be chopped out or repeated, and I have a short utility to automatically insert 'return to Basic' between each note of a M/C file and test the results.

The library is not complete as yet, I am aiming for 100 tunes as a starting point with just over 60 completed to date. Unfortunately I will not have the time to actually run the library when completed, so if someone out there is willing to take it on, then please get in touch, I must stress though that it is intended as a free service to Dragon users and not a commercial proposition (harities excepted!). Sorry, forgot to mention, not having a disc-drive, the library is cassette based with four tunes on each (C12) tape.

Anyone interested in taking the library, or helping to set it up. Please give me a call or drop me a line at home or work.

Stuart Beardwood
1 West Bottom
Hob Lane
Norland
Sowerby Bridge
W. Yorkshire HX6 3QL

PS I haven't a disc-drive, I don't need a printer, I can only program in Basic and I don't play adventure games. The new Dragon users mentioned in July's editorial could well be put off by thinking that DU's only catering for advanced users and leaving behind the beginners like me!!

EVERY single issue we have at least one person saying there was nothing interesting it and why don't we run the same things we ran in the issue before that — and one person saying it was the most interesting issue he's seen for ages. Please everyone? Of course we do — we just can't do it every month, that's all. I think Stuart's library idea is going to please quite a few people, though. Thanks from Dragoners for your efforts, and the efforts of the people who are supporting the school.

Club information

THANK you very much for publishing the fact that I am involved in forming a new user group. However the exact details you published were not correct. I feel that this was my fault as my initial letter was somewhat less than eloquent. I would then be thankful if you would publish the new details.

1) My postcode is fact NP4 5LN (the postman was not pleased).
2) The group is not only a local one but a postal group with newsletter, etc.(I will however be more than glad to arrange meetings in my area).
3) The first newsletter is almost ready (I will send you a copy).
4) Copies for prospective members can be obtained via me for a meagre 50p.
5) Subscription is £4/6mths or £7.50/year.
6) Members will have to be involved in the group.

I am very grateful for your help.

R. Ball, 75 Sunnybank Rd
Pontypool, Gwent NP4 5LN

HAH! Don't mention post people around here just at the moment. The west end posties are on strike again, and if there's anything essential missing from DU this month, that's why.

This is the chance to air your views — send your tips, compliments and complaints to Letters Page, *Dragon User*, 12-13 Little Newport Street, London WC2H 7PP.

Machine code wanted for beginners

I have owned a Dragon 32 for one and a bit years now and have been an avid reader of your magazine for the same period (since Feb 1986).

I have read one or two letters from other readers asking for the articles to the less basic (excuse pun) and in one case the plea was "less moronic".

May I plead the opposite case. Although I am of mature years, I have a computing age of probably five. I am slow by starting to understand Basic and can even modify one or two printed programs to the extent that they still work. BUT Machine code gives me a headache.

Would it be possible for someone to write an explanation in juvenile terms (computer-wise) so that I, and others like me, can at least enter Mnemonics via an assembler and get something out.

I have a complete set of 'INPUT' magazines, latest edition and have taped their Dragon assembler. I also have CO-RES from Microdeal. All I ever get is error messages. Even these flash across the screen too fast for comprehension. My hair was grey when I started but I shall start pulling it out quite soon.

If your staff authors consider it beneath them maybe one the readers who are regular correspondents could take up the challenge. I have two books by John Vander Riyden beside me now, and they don't help either. PLEASE, what am I doing wrong?

R.F. Harding
27 Down Amprey
Cirencester
Glos. GL7 5QS

Staff author? STAFF AUTHOR?? In this business, we think a staff author is someone who writes for Wizard's Weekly... **I wish we had one or two. At least then I would know where they were when I wanted them.**

Seriously, is anyone reading this interested in taking up the challenge? If so, drop the editor a line.

Hi Score Corner

ACCORDING to our postbag, Chuckie Egg **is one of the most popular games of all time. Why else do our readers spend hundreds of hours playing it? Paul lasikiewicz's challenge in DU June called forth the cream of the crop.**

WHILE reading your Hi-Score Corner, I came across Paul lasikiewiczs's high score for *Chuckie Egg*, which was 222,570 on level 22. My sister got onto level 22 as well but got a higher score of 236,580. She pleads for this letter to be printed. Please let me know if anyone should beat this as my sister wishes to strangle them.

Michael McCullough
129 Cregogh Road
Belfast
N. Ireland
BT6 0LA

THIS woman will go far. Not only has she talent, but she has a manager as well. Ever the persons to oblige, Dragon User **lines up the victims.**

REPLYING to Paul lasikiewicz in June's issue, the highest score I have ever got is 239,030 on level 21.

Simon Elencowe
3 Sandringham Road
Kings Sutton
Banbury
Oxon
OX17 3QS

SHORT and to the point. Dene Bebbington goes for philosophy as well.

IN response to Johnathan Baker's letter in the January issue, his high score on *Chuckie Egg* is not the highest. I have scored over 240,000 before level 20.

Unfortunately, the problem with high scores is that there's usually someone else who has managed to better your score. I imagine that scores between 300,000 and 400,000 on *Chuckie Egg* would be quite hard to achieve, or is there somebody who can prove me wrong?

Dene Bebbington
60 Quarry Clough
Stalybridge
Cheshire SK15 2RW

THE problem with high scores? I thought that was the whole point of high scores ... but see below.

REFERRING to the June issue Hi-Score Corner, I scored 250,610 on *Chuckie Egg* (level 24) and that was nearly three years ago! I would like to ask some questions that have been bothering me.

1) There aren't any proper instructions in *Wizard's Quest*. What do I do?

2) Whatever happened to a proposed game by Roy Coates called *Talisman*?

3) Are there any good programs for light pens?

4) I can't get anywhere in *Return of the Ring* (can't find breather mask, etc.) Help!

Graham Bruce
11 Garrick Road
Hartlepool
Cleveland
TS25 4EJ

1) Write to Microdeal, or Pete Gerrard, or both. 2) Because Roy's programmer suddenly realised he was starving and went and got a proper job 3) See last month's news page 4) someone help this chap. By the way, we are trying to sort out your missing tape query, Mr. Bruce. If at first you don't succeeed, keep on swinging, as the spider said. But on to the next victim:

I WAS reading June's Hi-Score Corner and I found a score on Chuckie Egg that was 222,570 on level 22. My highest score was 251,660 on level 23.

Robert Hough
364 Vicarage Lane
Marton
Blackpool
Lancs
4Y4 4ND

WHILE reading June's issue, I came across a letter from Paul lasikiewicz who said his hi-score for Chuckie Egg was 222,570. My hi-score for Chuckie Egg is 326,650, level 27!

Some other hi-scores:
Module Man 156,650
Hungry Horace 6,794,320
Mr. Dig 2,429,290

If anyone needs help on *Trekboer*, *Juxtaposition* or *Shenanigans*, please write to me. I also would be very glad if anyone could help me with *The Vortex Factor* (from year 3032(2)BC).

John Korsvold
The Dragon Devil
2824 Redalen
Norway

SEE, Dene? And further ...

PLEASE find enclosed my cheque for £14, and my payment slip. Keep up the good work.

Carl Truett
11 Hayden Road
Rushden
Northants NN10 0HX

PS My high score for *Chuckie Egg* is 413,940 at level 32. The normal ducks go much faster and there are two mad ducks. If anyone can do level 32, then they eally know where their towel is. My other good high scores are 316,600 on *Donkey King*, 625,480 on *Mr. Dig*, 456,800 on *Module Man* and 1,335,078 on *Hungry Horace*. (None of them POKE aided.)

NOW that's the kind of letter we really **like. Where can we go from there? Time for another poem, obviously.**

I thought I would write and tell you the game we have, *Moon Cresta*, is very good. The highest score is 25,000, I'm afraid I can only reach 23,000, so out of frustration I wrote a poem about it.

MOON CRESTA what a game
Score 25,000 and you reach fame.
It's good fun yet frustrating.
I guess its all the waiting.
What makes the game so rewarding
Is, without any warning
FAR OUT, FAR OUT
What satisfaction.
It causes an adrenaline reaction.
Moon Cresta is my game
Because in the end I
Type in my name.
Moon Cresta!

Mrs. V. Benton
120 Bentley Court
Moor Street
Luton
Beds. LU1 1EZ

FAME at last, and she didn't even have play *Chuckie Egg* **to do it.**

If you have any new products for the Dragon — software or hardware — ring the News Desk on 01-437 4343

Trojan Heroes

DRAGON USER wishes to apologise to Trojan Micro Computer Software and Accessories for describing the original Trojan Light Pen software as 'disappointing' in the August issue.

This was entirely due to a misunderstanding about Trojan's reasons for permitting the software to be updated. Says Trojan's Tom Jones, who was extremely good natured about our clanger, "The Trojan lightpen software has sold over 20,000 copies overall, and was very well reviewed in *Dragon User*. We're still very happy with it. We have given informal permission for alterations to be made to it, within our copyright, as long as it is checked by us."

Trojan can be contacted on 0792 205491 if anyone has any questions.

Good show

HERE are a couple of corrections to recent news items:

The North of England Dragon Show and Convention will be taking place as and where we said it would, with the exception that the venue is the **Bishop Henshaw Upper School**, Rochdale, and not the Bishop Hanslow Upper School as we claimed.

Sorry, Vicar!

"Do tell people to come along" says Pulser's Brain O'Connor. "Some dealers will give up shows if they can't get a reasonable attendance and cover their costs. This will be a bit different from the average Dragon show."

It's your show. Go, go, go. We are.

Also apologies to Hans Christian Andersen Software. Of course their program *Superwriter II* can't be 'loaded into memory and then saved and loaded onto another disc' as we claimed in the July *Newsdesk*. What we meant was that the *data* files can be saved to another disc.

Sorry, Hans!

NDUG

HERE I am, clutching a June issue of *Dragon Update* bearing the inscription 'news for September', and it's nearly the end of July. It's an arrangement known as the 'double time scheme' to drama critics, and 'darned confusing' to *Dragon User*.

Anyway, here are the contents of June's *Dragon Update*: test run of the US J & R Electronics Bankers' Board design from *The Rainbow*, utility reviews, easy machine code, OS-9 for beginners, *Telewriter* and SuperDOS, a joystick/cursor key mod, some bits on DeltaDOS, a couple of graphics routines, some screen dump samples (and a drawing competition), the usual small ads and a suggestion that DU and *Update* should rule the world, seeing as how some people seem to think we do.

Well, why not, Paul? As long as I get to wear a crown.

Update also warns that business advertisers will have to contribute to costs in future, or risk having their ads 'bumped'.

DU thinks this is fair enough. Although *Update* is not strictly a commercial organisation, they are still in business insomuchas they have to pay out money for printing etc. just as we do, and need contributions from people using their services.

The National Dragon Users Group can be contacted at 6, Navarino Road, Worthing, Sussex. Phone: 0903 207585.

Can't stand printers?

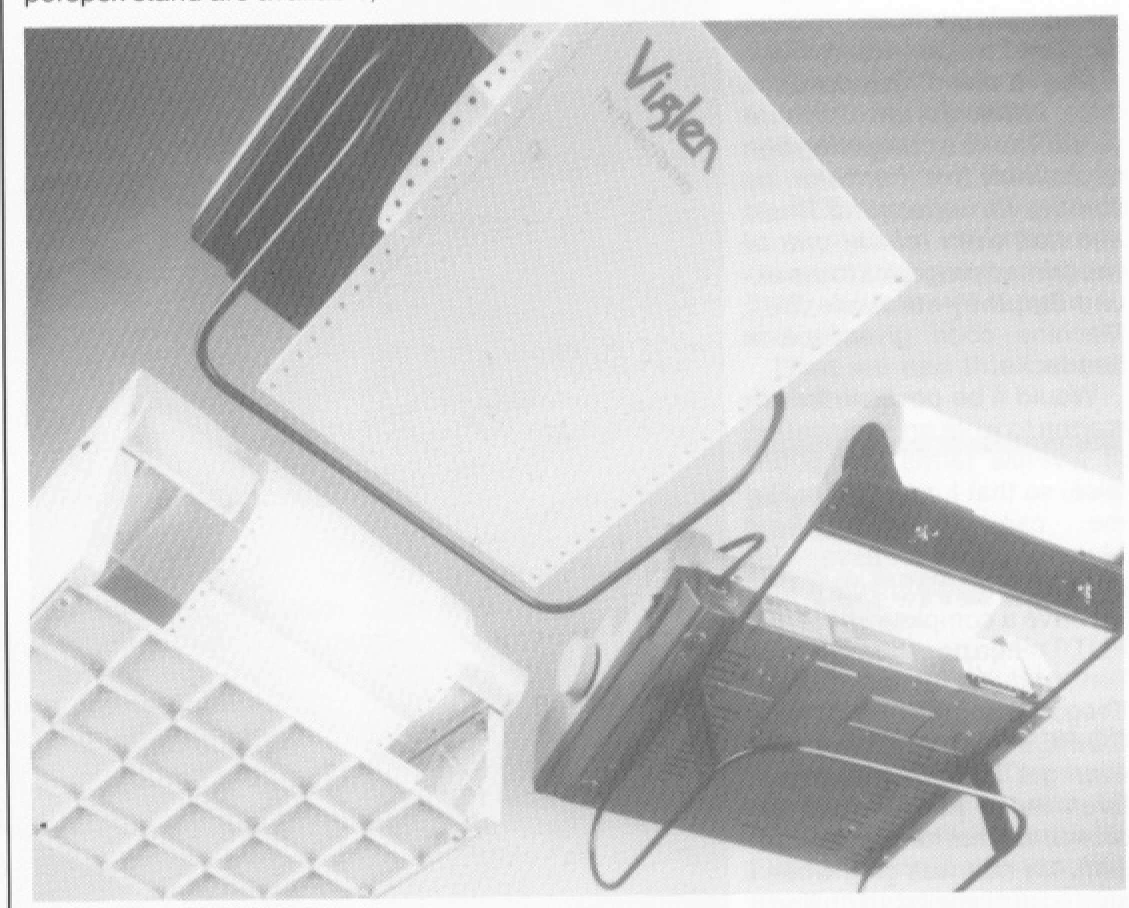
VIGLEN has announced a new range of printer stands for use with most 80 or 136 column printers.

At the bottom of the range is a metal stand finished in black, designed for use with the Centronics GLP II printer. There is a larger version of this stand with a paper roll holder.

Two models of smoky or clear look perspex stand are available, with room for continuous stationery underneath.

Viglen also supplies a high-impact plastic printer stand, colour co-ordinated to the usual printer colour, with room for 1,000 sheets of continuous stationery.

Prices range from a very affordable £9.95 up to £29.95. See your computer dealers, or contact Viglen on 01 843 9903 for a list of dealers.



Correction

B. Yeoman Walker has written to say: 'the printers have truncated more than the title in my article. Several lines are missing from the bottom of Figure 1 on page 19 of the July issue, which will no doubt cause some readers to wonder what has happened. (*Not just readers — Ed.*) Here are the missing notes:

1. There is no Elite facility.

2. There are no ESC P codes.

3. The ENGLARGED mode is only available by SO (= Shift Out) .CHR$(14).

4. The CONDENSED mode is only available by SI (= Shift In). CHR$(15).

5. There is no Italic facility (ie No ESC 4 or ESC 5, or alternative).

New software for review should be sent to *Dragon User* 12-13 Little Newport Street, London WC2H 7PP.

Dragon guide book

Program: *Magbase*
Supplier: Pulser Software, 36 Foxhill, High Crompton, Shaw, Oldham, Lancs OL2 7NQ
Price: various according to versions

THERE have been surprisingly few database programs written for the Dragon since the late arrival of the disc system, and most of those packages that did reach the shelves (can you remember buying Dragon software from a shop?) failed miserably due either poor programming or weak program design. Having got myself a reasonable database program what do you use it for? Computers are ideally suited for sorting and searching through large amounts of information but what kind of information do most home computers have that requires a computer to sort it out for them? Why, keeping track of the wonderfully witty and informative articles in our beloved *Dragon User* magazine I hear you both cry!

Pulser Software have now released a database program which has been specifically written for indexing magazine articles and comes either on its own or with a choice of data files that contain an index of the articles from within the covers of such esteemed journals as *Dragon User* and *Dragon Update.* The program is available on cassette and disc versions are available for the usual systems including DragonDOS, DeltaDOS and CumanaDOS.

The program is written in Basic which is reasonable for small database situations and has been around long enough that there are no 'bugs' in it. Records may be searched for using either a 'target' string or range of record numbers and output, as you would expect, may be directed to either the screen or the printer. The printer routines are, by default, set up for Seikosha or Tandy printers but this may be re-configured to suit your own system. All the other usual database function are provided to allow sorting, record modification and addition etc. etc.

Trying to remember which issue a particular article was in is virtually impossible, *Magbase* will not only find it for you but will point out other articles on the same subject. *Magbase* must be worth its asking price for the *Dragon User* index alone without the database software.The only real snag with the *Magbase* program is its elegance. Having selected and confirmed an option it is annoying to be asked to 'press a key to start'. It's a small gripe I know, but when you use the program for more than a few minutes it becomes very annoying.

Roy Coates



Success for the search

Program: *Data Retrieval System, D.R.S.*
Supplier: Grosvenor Software, 2 Beacon Close, Seaford, E.Sussex BN25 2JZ
Price: £9.95

THE introductory screen has a 1986 copyright message, but on the main menu the date is 1983. This, coupled with the title, leads me to suspect that this program shares a common ancestry with the *Salamander DRS* System which was around in the early days of the Dragon. Never mind the history, what does it do and how well does it do it? Well basically it is a database system and it works extremely well.

The program is supplied on cassette, but the manual gives instructions for transferring to disc (CLEAR 100 first if transferring to disc — that's been omitted from the manual!). In its cassette form the program copes with 22,000 characters of data, all held in memory at once. A disc file can be as large as 154,000 characters and is treated as up to 7 segments of 22,000 characters each. Data is stored as a number of records, each of which can contain up to 480 characters split into 36 fields. Data is input and processed via very flexible user definable screen layouts, one record displayed at a time with full screen editing facilities. The program is written in machine code and works very fast.

So how do you use the program? Having loaded it, a menu is displayed suggesting you press D to design your screen layout. You then set up the format of the screen and the number, size and type of each field in the record; you can also add titles to the fields. The photograph shows a typical format screen, defining 15 fields (A-I are alphanumeric and 1-5 are numeric). For example, the format for "BIRTHDAY" defines that up to three entries may be made consisting of a two digit number for the date, a three character month and a four digit year. Having completed the screen layout this is saved and you return to the main menu. You now start adding data by selecting the 'A' command. Data can only be entered into the defined fields and must be of the correct type, ie only numbers entered into a numeric field. As each field is filled the next is selected by using either the ⟨ENTER⟩ key or the cursor arrow keys. You do not need to fill all fields, the program copes quite happily with empty ones. When a record is complete, you press ⟨BREAK⟩ and are prompted by 'Add' to press ⟨ENTER⟩, you are then prompted by "OK,Next/Menu N" so press ⟨ENTER⟩ yet again and you are ready to start filling the next record. I found this tedious, but it works and gives maximum flexibility. The second photograph shows a typical datarecord. You can continue entering data until you have completed a full list of your friends, their addresses, children's ages or whatever.

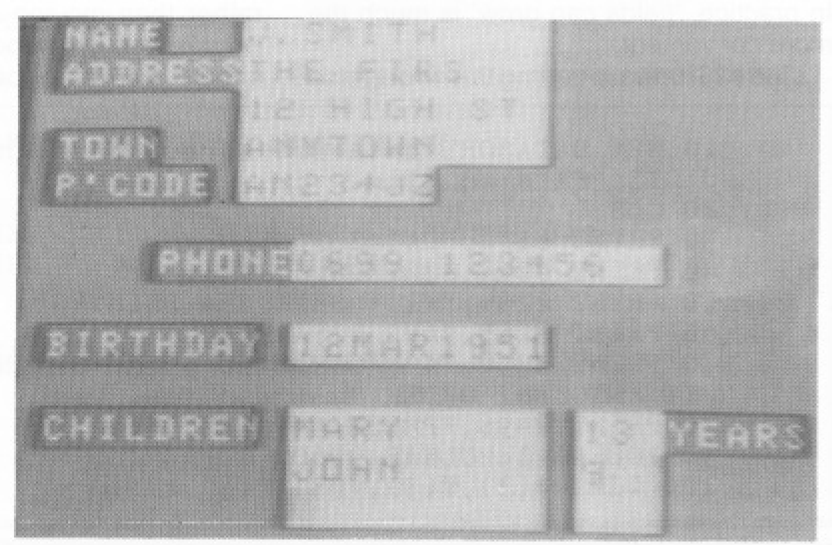
Having got your database, what can you do with it? The first thing is to save it on tape or disc in case of power failure! Then you can reformat it if you don't like the original screen design, update records, print it out either in total or selectively, but most importantly you can do searches on it. You enter 'S' from the main menu and you are presented with a blank version of your original screen design. Into this you can enter search characters in any position, a range for the initial character of a field and ranges for numerical values, eg ⟨10; the program then ANDs these all together and does a search for all records that meet all these criteria, eg you could do a search for all people whose surnames begin with a letter between J and P, who live in Northampton and have a child less than 5 years old; or look for all music by Beethoven recorded by the LSO in 1984.

Just a few minor criticisms of the program: it's written for the Dragon 32 and can't take advantage of the extra storage provided by the D64. As with a great many Dragon programs, you are committted to upper case letters only. It would have been nice if they had added a lower case driver using the graphics screen (although this would reduce storage capacity), and three key strokes to enter each new data record really is a bit tedious. Finally, if you are not lucky enough to have a disc drive, I don't think its really worth bothering to try and do anything serious with any database program using cassette storage, it's just too slow, you could have looked up a written version of the information in half the time.

To sum up, if you've got a Dragon with a disc drive, and you've not got Flex or OS9, and you want to make serious use of it, then this really is a must for you. It's a piece of professional software at an amateur price.

Bernice Hennessy





Datasort

***Mike Hosken** presents a database program which just grew . . .*

THIS is a long program, because it does so many useful jobs. But it's not as dreadful to type in as some of your more major machine code listings. It's not particularly well structured; it might even be accused of suffering from the spaghetti syndrome. The reason is that the program has gradually evolved, with extra facilities being added as my needs expanded. And I simply daren't re-structure it for fear of bugs.

There is quite a lot to say, so I will do it in note form.

Description:
This is a program of approximately 12K which will accept any form of 'tabulatable' data; inspect, alter or delete headings or records; sort numerically or alphabetically on any field; save/load using DragonDOS including an option to combine two sorted databases into a sorted master database; compile an index by optional combination of identical entries; print a customised tabulation, either a whole database or by use of up to three simultaneous selector keys; total and average numberical data, all or selected.

Example:
Table 1 gives details of variables in *Datasort*, compiled on Datasort. (Note that sorting is operational on both fields: second column entries are alphabetical within the first column sorted sequence.)

Hardware:
Datasort will run on any Dragon but CLEAR, PEEKs and POKEs are for Dragon 32. Examples printed here were done on a daisywheel printer but program listing CHR$ numbers are for the Seikosha dot matrix printer using continuous stationery.

Program and Operational Notes:
Line 240: if your printer does not have automatic line feed on using the last character then the full printer width can be used.

Line 250: not possible on other kinds of database programs, where fields have to be specified before you start, regardless. In practice, 'fields can grow' is much the more convenient.

Line 420: beware getting this wrong else sorting will treat 9 as greater than 100!

Line 500: invaluable if you are compiling an index, to a book perhaps — or set of *Dragon Users*.

Subroutine 630 to 650 produces a marker to show present maximum field width.

975 DIR can be inserted if you would like to have your files listed on screen before making a choice.

Line 1000 sends you back for another try if you don't specify a proper file title: it may be safer to exit back to the menu with THEN 1210, if you prefer.

Data could be lost if the same happened at line 1410: so again, THEN 1210 (instead of 1390).

Line 1440: to make use of this facility (which saves both time and memory under many circumstances), process database A, sort it, and save to disc. Re-RUN with database B, and sort it. Reload database A, effectively 'on top of' B. (Re-sorting at this stage would, if the memory is now almost full, take a very long time.) Save onto disc, which can be done as a sorted combination AB. Re-RUN, Loading AB for printing, modification, arithmetic, etc.

Line 1840: sorting usually uses the fastest swapping procedure but this can be overcome here. Eg, sort book titles fast, then sort by authors but 'retain previous sequences': thus Dickens's novels will be in alphabetical order but all before Shaw's plays (in alphabetical order).

Line 2280: any headings are printed to the left margin so only exceptionally should spaces be inserted to the left of the first field.

Line 2390: CHR$ (14) puts the printer into enhanced mode and CHR$(15) returns it to normal.

Line 2460: tabulations are easier to read in blocks of five lines.

Line 2650: using INKEY$ rather than INPUT makes it easier to browse records using N option. For other options don't press ENTER until AFTER selecting the record number. (Or change to INPUT if preferred.)

With the Delete option the gap in the array is filled by bringing up the last entry rather than moving ALL records up one (which would be too slow). So deleting should be done before sorting where possible. Also, any systematic editing should start at the end and work upwards, ideally.

Line 2830: any search goes right through the database, rather than stopping after finding one appropriate entry.

I hope line 2970 is clear enough. If the displayed field entry is alright then just press ENTER to retain it. If it's NOT alright then type in everything that you want in that field (including what was in it previously if you merely want to extend it) and THEN press ENTER.

Line 3650 would, on a database of second-hand cars, pick out all/any 'Ford' models, and line 3660 all the injected versions if the model number ends in 'i'. Line 3720 enables it to select the ones priced below, say, 3500, but on a members' list it could check all those with names starting with S or earlier in the alphabet.

Line 4180 is simply an anti-abortion safety device. But it is so important that it demands a three-key response whereas the usual ones have been the two-key OK or the single-key Y or N: if your universal preference is for Y/N then program all such choices in whichever way you like.

Incidentally, *Datasort* files can be loaded into a word processor or perhaps into a label-printing utility program to add even greater flexibility to the whole system. But if you've already got some data and you want to convert it into *Datasort* format you need simply to build a converter program. This can be done if you CHAIN the loading section of your own program, with a small middle section if any processing is necessary (such as the addition of *Datasort* flags as listed in Table 1), and the 'Save data on disk' sector of *Datasort*.

Personal Note:
Not my prettiest program: it grew a bit haphazardly, but it is much my most useful one (even if accounts may actually be more important).

I have another version for tape storage, and I would be interested to hear if anyone is interested in that.

(Mike Hosken is not an easy chap to get hold of these days, but if anyone is interested in alternative versions, drop us a line, and we will see if we can get in touch with him eventually.)

```
10 REM DATASORT database sort, store and tabulate: SEIKOSHA printer an
d DISK file edition.
20 CLS
30 PRINT @172,"DATASORT";
40 PRINT @204,STRING$(8,243);
50 PRINT @229,"WITH SEIKOSHA PRINTER"
60 PRINT @265,"AND DISK FILES"
70 PRINT @352,"A PROGRAM TO STORE STRING AND/ORNUMERICAL DATA AND SORT
 IT INTO ANY ORDER."
80 PRINT @496,"PRESS ANY KEY..";
90 PCLEAR 1: CLEAR 9000
100 DIM G$(3),T(3),V(3),Y(3)
```

```
110 IF INKEY$="" THEN 110
120 CLS: PRINT @96,"DO YOU WISH TO RE-ENTER OLD DATAFROM DISK?  (Y/N)"
130 INPUT D$
140 IF D$<>"Y" AND D$<>"N" THEN 130
150 CLS: IF D$="Y" THEN 970
160 REM >>>>>>>>>> setting up new records
170 PRINT @96,"INPUT ANY TITLE OR LIST-HEADING FOR THE WHOLE SYSTEM
        (OR JUST 'ENTER' BY ITSELF)."
180 LINE INPUT T$: CLS
190 PRINT @96,"HOW MANY items OF DATA (FIELDS) MAKE UP EACH LINE (RECO
RD)";
200 INPUT B: B=B-1: CLS
210 IF B<1 THEN CLS0: GOTO 190
220 A=1200/(B+1)
230 DIM F$(A,B),C(B),H$(B),R(B),SL(B)
240 PRINT "FOR PRINTING AT ONE LINE PER    RECORD PLEASE PLAN AN ALLOC
ATIONOF SPACES (AND PRINT GAPS) TO   TOTAL A MAXIMUM OF 79.": PRINT
250 PRINT "1 = PRESET FIELD WIDTHS         0 = FIELDS CAN GROW": PRINT
260 INPUT "CHOOSE";U: CLS
270 IF U<>1 AND U<>0 THEN 250
280 PRINT "INPUT ANY FIELD TITLES/HEADINGS."
290 FOR L=0 TO B
300 PRINT "FIELD"L+1;
310 LINE INPUT F$(0,L)
320 SL(L)=LEN(F$(0,L))
330 IF U=0 THEN 360
340 INPUT "FIELD WIDTH";SL(L)
350 IF LEN(F$(0,L))>SL(L) THEN PRINT "HEADING TOO WIDE!": GOTO 300
360 NEXT L
370 CLS
380 FOR L=0 TO B: PRINT F$(0,L): IF U=1 THEN PRINT "WIDTH"SL(L)
390 NEXT L
400 INPUT "TYPE 'OK' OR 'NO'...";A$: CLS
410 IF A$<>"OK" THEN 280
420 PRINT "INDICATE WHICH TYPE OF DATA IS  TO BE DEALT WITH IN EACH FI
ELD: N = NUMBERS OR L = LETTERS ETC.": PRINT
430 FOR L=0 TO B
440 PRINT "FIELD"L+1;F$(0,L)" ";
450 INPUT A$
460 IF A$<>"N" AND A$<>"L" THEN 450
470 IF A$="L" THEN C(L)=1 ELSE C(L)=0
480 NEXT L
490 INPUT "TYPE 'OK' OR 'NO'...";A$: CLS: IF A$<>"OK" THEN 420
500 PRINT "SHOULD RECORDS WITH IDENTICAL": PRINT F$(0,0)
510 INPUT "ENTRIES BE COMBINED?  (Y/N)";A$
520 IF A$<>"Y" AND A$<>"N" THEN 500
530 IF A$="Y" THEN CR=1
540 CLS: PRINT "NOW IS THE TIME TO START PUTTINGIN ALL THE DATA."
550 REM >>>>>>>>>> data input
560 PRINT "USE @@@ AS A DUMMY ENTRY TO     END DATA INPUT."
570 R=R+1: IF R>A THEN PRINT "STORAGE ARRAYS NOW FULL.": GOTO 1230
580 PRINT "RECORD NUMBER";R
590 PRINT STRING$(13+LEN(STR$(R)),131)
600 FOR L=0 TO B
610 IF F$(0,L)<>"" THEN PRINT F$(0,L) ELSE PRINT "DATA FOR FIELD"L+1
620 GOSUB 630: GOTO 660
630 X=PEEK(136)*256+PEEK(137)
640 IF X+SL(L)>1535 THEN PRINT @511: PRINT @X-1056,;: GOTO 630
650 POKE X+SL(L),191: RETURN
660 LINE INPUT F$(R,L)
670 IF F$(R,L)="@@@" THEN R=R-1: GOTO 1210
680 IF L>0 OR R=1 OR CR=0 THEN 740
690 KK=0
700 FOR K=1 TO R-1
710 IF F$(R,L)=F$(K,L) THEN PRINT "AMEND RECORD"K: M=K: KK=K
720 NEXT K
730 IF KK>0 THEN R=R-1: GOTO 2970
740 IF LEN(F$(R,L))>SL(L) AND U=0 THEN SL(L)=LEN(F$(R,L)): GOSUB 840
750 IF LEN(F$(R,L))>SL(L) AND U=1 THEN F$(R,L)=LEFT$(F$(R,L),SL(L))
760 NEXT L
770 CLS
780 PRINT PEEK(35)*256+PEEK(36)-PEEK(33)*256-PEEK(34)"memory free"
790 PRINT @21,"@@@ to stop";
800 L=R: GOSUB 3090
810 PRINT
```

```
820 GOTO 570
830 REM >>>>> check line length
840 W=0: S=0
850 FOR LL=0 TO B
860 W=W+SL(LL)+R(LL): IF R(LL)>0 THEN S=1
870 NEXT LL
880 PRINT "(LINE LENGTH"W;
890 IF S=1 THEN PRINT "INCL. GAPS.)" ELSE PRINT "WITH NO GAPS.)"
900 IF J=6 THEN 930 ELSE IF J=3 THEN RETURN
910 FOR LL=1 TO 500: NEXT LL
920 RETURN
930 FOR LL=0 TO B
940 PRINT "FIELD"LL+1,SL(LL)"CHARACTERS"
950 NEXT LL
960 IF INKEY$="" THEN 960 ELSE 1210
970 REM >>>>>>>>>> reading data from disk
980 PRINT @96,"INPUT THE NAME OF THE REQUIRED  DATA FILE."
990 INPUT H$
1000 IF H$="" THEN 980
1010 CLS
1020 PRINT @160,"CHECK THAT THE DISK IS IN PLACE.PRESS ANY KEY WHEN RE
ADY...."
1030 IF INKEY$="" THEN 1030
1040 CLS: PRINT @236,H$
1050 FREAD H$;A
1060 FREAD H$;B
1070 FREAD H$;CR
1080 FREAD H$;Y
1090 FREAD H$;U
1100 FLREAD H$;T$
1110 IF R=0 THEN DIM C(B),F$(A,B),H$(B),R(B),SL(B)
1120 FOR L=0 TO B: FREAD H$;C(L): FREAD H$;Z: IF Z>SL(L) THEN SL(L)=Z
1130 NEXT L
1140 FOR L=0 TO B: FLREAD H$;F$(0,L): NEXT L
1150 FOR L=R+1 TO R+Y
1160 FOR LL=0 TO B
1170 FLREAD H$;F$(L,LL): PRINT F$(L,LL)
1180 NEXT LL,L
1190 CLOSE
1200 R=R+Y: IF R=Y THEN Y=0
1210 REM >>>>>>>>>> menu
1220 CLS1
1230 PRINT @36,"main menu"
1240 PRINT @96,"1 = SAVE DATA ONTO DISK"
1250 PRINT "2 = SORT DATA INTO A NEW ORDER"
1260 PRINT "3 = PRINT"
1270 PRINT "4 = INSPECT/ALTER/DELETE DATA"
1280 PRINT "5 = ADD MORE DATA"
1290 PRINT "6 = CHECK LINE LENGTH"
1300 PRINT "7 = ARITHMETIC"
1310 PRINT "8 = LOAD EXTRA DATA FROM DISK"
1320 PRINT "9 = CHANGE MAIN TITLE": PRINT
1330 PRINT "0 = FINISHED": PRINT
1340 INPUT "    WHICH NUMBER";J: CLS
1350 IF J>9 THEN CLS0: GOTO 1230
1360 IF J=0 THEN 4170
1370 ON J GOTO 1380,1710,2100,2620,780,840,3120,970,4100
1380 REM >>>>>>>>>> saving data onto disk
1390 PRINT "INPUT AN 8-CHARACTER FILE NAME -";
1400 INPUT H$
1410 IF H$="" OR LEN(H$)>8 THEN 1390
1420 IF Y=0 THEN Q=0: GOTO 1490 ELSE Q=1
1430 PRINT "IF YOUR TWO DATA FILES ARE asciiSORTED THEY CAN BE SAVED A
S A   COMBINED-SORTED FILE."
1440 INPUT "DO YOU WISH THIS TO BE DONE?    (Y/N)";A$
1450 IF A$<>"Y" AND A$<>"N" THEN 1430
1460 IF A$="N" THEN Q=0: GOTO 1490
1470 INPUT "WHICH FIELD NUMBER FOR SORTING? ";Z: Z=Z-1
1480 IF Z<0 OR Z>B THEN CLS0: GOTO 1470
1490 PRINT: PRINT "CHECK THAT THE DISK IS IN PLACE.PRESS ANY KEY WHEN
READY...."
1500 IF INKEY$="" THEN 1500
1510 CLS
1520 M=1: N=R-Y+1
1530 PRINT @224,"SAVING YOUR ";H$;" DATA FILE."
```

```
1540 FWRITE H$;A
1550 FWRITE H$;B
1560 FWRITE H$;CR
1570 FWRITE H$;R
1580 FWRITE H$;U
1590 FWRITE H$;T$
1600 FOR L=0 TO B: FWRITE H$;C(L): FWRITE H$;SL(L): NEXT L
1610 FOR L=0 TO B: FWRITE H$;F$(0,L): NEXT L
1620 FOR L=1 TO R-1
1630 IF Q=0 THEN S=L: GOTO 1650
1640 IF F$(M,Z)<F$(N,Z) OR N>R THEN S=M: M=M+1 ELSE S=N: N=N+1
1650 FOR LL=0 TO B
1660 FWRITE H$;F$(S,LL): PRINT F$(S,LL)
1670 NEXT LL,L
1680 IF Q=0 OR N=R THEN S=R ELSE S=M
1690 FOR L=0 TO B: FWRITE H$;F$(S,L): PRINT F$(S,L): NEXT L
1700 CLOSE: GOTO 1210
1710 REM >>>>>>>>>> sorting procedure
1720 FOR L=0 TO B
1730 PRINT "FIELD"L+1"IS ";
1740 IF C(L)=1 THEN PRINT "ALPHABETICAL" ELSE PRINT "NUMERICAL"
1750 IF F$(0,L)<>"" THEN PRINT F$(0,L)
1760 NEXT L
1770 INPUT "WHICH FIELD IS TO BE SORTED";K: K=K-1
1780 IF K<0 OR K>B THEN CLS0: GOTO 1710
1790 IF C(K)=1 THEN 1840
1800 PRINT: PRINT "CHOOSE THE REQUIRED SORT:-": PRINT
1810 PRINT "1 = DECREASING (TOP FIRST)      0 = INCREASING"
1820 INPUT M
1830 IF M<>1 AND M<>0 THEN CLS0: GOTO 1800
1840 PRINT "CHOOSE AGAIN:-"
1850 PRINT "1 = FASTER"
1860 PRINT "0 = RETAIN PREVIOUS SEQUENCES"
1870 INPUT H: CLS
1880 IF H<>1 AND H<>0 THEN CLS0: GOTO 1840
1890 FOR L=1 TO R
1900 N=L
1910 IF C(K)=1 THEN 1920 ELSE IF M=1 THEN 1950 ELSE 1980
1920 FOR LL=N TO R: IF F$(LL,K)<F$(N,K) THEN N=LL
1930 NEXT LL
1940 GOTO 2000
1950 FOR LL=N TO R: IF VAL(F$(LL,K))>VAL(F$(N,K)) THEN N=LL
1960 NEXT LL
1970 GOTO 2000
1980 FOR LL=N TO R: IF VAL(F$(LL,K))<VAL(F$(N,K)) THEN N=LL
1990 NEXT LL
2000 IF N=L THEN FOR LL=0 TO B: PRINT F$(L,LL): NEXT LL: GOTO 2030
2010 IF H=0 THEN 2050
2020 FOR LL=0 TO B: Z$=F$(N,LL): F$(N,LL)=F$(L,LL): F$(L,LL)=Z$: PRINT
 F$(L,LL): NEXT LL
2030 NEXT L
2040 GOTO 1210
2050 FOR LL=0 TO B: H$(LL)=F$(N,LL): NEXT LL
2060 FOR LL=N-1 TO L STEP -1
2070 FOR LS=0 TO B: F$(LL+1,LS)=F$(LL,LS): NEXT LS,LL
2080 FOR LL=0 TO B: F$(L,LL)=H$(LL): PRINT H$(LL): NEXT LL
2090 GOTO 2030
2100 REM >>>>>>>>>> printing
2110 CLS
2120 PRINT @9," PRINTING  MENU ";
2130 PRINT @64,"1 = ONE NUMBERED RECORD"
2140 PRINT "2 = SELECTED RECORDS TABULATED"
2150 PRINT "3 = ALL RECORDS TABULATED": PRINT
2160 PRINT "4 = RETURN TO MAIN MENU"
2170 PRINT @352,"INPUT YOUR CHOICE";
2180 INPUT M: CLS
2190 IF M<1 OR M>4 THEN CLS0: GOTO 2120
2200 IF M=4 THEN 1210 ELSE GOSUB 830
2210 PRINT "FIXING THE REQUIRED TABULATION  PARAMETERS:-": PRINT
2220 IF R(1)=0 THEN 2260
2230 PRINT "IS THE PREVIOUS TABULATION      LAYOUT ALRIGHT?"
2240 INPUT "TYPE 'OK' OR 'NO'...";A$: CLS
2250 IF A$="OK" THEN 2320
2260 FOR L=0 TO B
2270 PRINT "FIELD"L+1;F$(0,L)
```

```
2280 INPUT "HOW MANY SPACES TO THE LEFT OF  THE FIELD ENTRY";R(L)
2290 IF R(L)<0 OR R(L)>80 THEN PRINT "DON'T BE DAFT!": GOTO 2280
2300 PRINT: NEXT L
2310 CLS
2320 PRINT @160,"PLEASE CHECK THAT THE PRINTER ISREADY, AND THEN PRESS
 'ENTER'."
2330 INPUT A$
2340 PRINT "THE PRINTER STILL ISN'T READY,  FOR SOME REASON.   PLEASE
CHECK AND REMEDY."
2350 PRINT #-2: CLS
2360 IF M=1 THEN GOSUB 2570: GOSUB 2510: GOTO 2100
2370 IF M=2 THEN 3440
2380 PRINT @229,"output to the printer"
2390 IF LEN(T$)>1 THEN PRINT #-2,CHR$(14)T$: PRINT #-2,STRING$(LEN(T$)
,"=")CHR$(15)
2400 IF LEN(F$(0,0))+LEN(F$(0,1))=0 THEN PRINT #-2: GOTO 2440
2410 PRINT #-2
2420 L=0: GOSUB 2510
2430 PRINT #-2
2440 FOR L=1 TO R
2450 GOSUB 2510
2460 IF L=INT(L/5)*5 AND L<R THEN PRINT #-2
2470 NEXT L
2480 PRINT #-2
2490 GOTO 2100
2500 REM >>>>> tabulating a record on the printer
2510 K=0
2520 FOR LL=0 TO B
2530 K=K+R(LL): PRINT #-2,TAB(K)F$(L,LL);: K=K+SL(LL)
2540 NEXT LL
2550 PRINT #-2: RETURN
2560 REM >>>>> choosing record number
2570 PRINT @160,"INPUT THE number OF THE RECORD  WHICH IS TO BE LISTED
 ON THE    PRINTER (OR '0' IF NOT KNOWN, TOBE SEARCHED FOR)."
2580 INPUT L
2590 IF L=0 THEN CLS: GOTO 2830
2600 IF L>R THEN CLS0: PRINT "THERE AREN'T THAT MANY ENTRIES!": GOTO 2
570
2610 CLS: RETURN
2620 REM >>>>>>>>>> inspecting/altering data
2630 PRINT "PRESS -"
2640 PRINT "aLTER dELETE iNSPECT nEXT  mENU";
2650 D$=INKEY$
2660 IF D$<>"A" AND D$<>"D" AND D$<>"I" AND D$<>"M" AND D$<>"N" THEN 2
650
2670 CLS
2680 IF D$="M" THEN 1210
2690 IF D$="N" THEN L=L+1: IF L=<R THEN 2770 ELSE 1210
2700 PRINT "INPUT THE number OF THE ENTRY OR'S' FOR A SEARCH."
2710 INPUT P$
2720 IF VAL(P$)>R THEN PRINT "THERE AREN'T THAT MANY ENTRIES!": GOTO 2
640
2730 IF VAL(P$)<0 AND P$<>"S" THEN 2700
2740 IF P$="S" THEN 2830
2750 CLS: IF D$="A" THEN 2960
2760 L=VAL(P$)
2770 GOSUB 3090
2780 IF D$="I" OR D$="N" THEN 2640
2790 INPUT "TYPE 'GO' TO DELETE             (OR 'KEEP' TO RETAIN)...";
D$
2800 IF D$<>"GO" THEN CLS: GOSUB 3090: PRINT: GOTO 2640
2810 FOR LL=0 TO B: F$(L,LL)=F$(R,LL): NEXT LL
2820 R=R-1: CLS: GOTO 2640
2830 PRINT "WHICH FIELD IS TO BE SEARCHED?  (1 TO"B+1")"
2840 INPUT N: N=N-1
2850 IF N<0 OR N>B THEN 2830
2860 PRINT "NOW INPUT THE ";
2870 IF C(N)=0 THEN PRINT "NUMERICAL ";
2880 PRINT "DATA TO BE FOUND."
2890 LINE INPUT M$
2900 CLS: Q=0
2910 FOR L=1 TO R
2920 IF M$=F$(L,N) THEN GOSUB 3090
2930 NEXT L
2940 IF Q=0 THEN PRINT "searched but with nil result"
```

```
2950 GOTO 2640
2960 M=VAL(P$)
2970 PRINT "INPUT NEW DATA OR 'ENTER' ONLY  TO RETAIN PRESENT FIELD EN
TRY."
2980 FOR L=0 TO B
2990 PRINT F$(M,L)
3000 GOSUB 630
3010 LINE INPUT D$
3020 IF D$<>"" THEN F$(M,L)=D$: IF LEN(F$(M,L))>SL(L) AND U=0 THEN SL(
L)=LEN(F$(M,L)): GOSUB 830
3030 IF U=1 AND LEN(F$(M,L))>SL(L) THEN F$(M,L)=LEFT$(F$(M,L),SL(L))
3040 NEXT L
3050 CLS: L=M: GOSUB 3090
3060 IF J<>4 THEN 550
3070 GOTO 2640
3080 REM >>>>> listing a record on VDU.
3090 PRINT "RECORD NUMBER";L
3100 FOR LL=0 TO B: PRINT F$(L,LL): NEXT LL
3110 Q=1: RETURN
3120 REM >>>>>>>>>> arithmetic functions
3130 PRINT @64,"YOU CAN HAVE TOTALS AND AVERAGESFOR ANY NUMERICAL DATA
, EITHER  FROM ALL OR SELECTED RECORDS.": PRINT
3140 PRINT "THE NUMERICAL FIELDS ARE:"
3150 FOR L=0 TO B
3160 IF C(L)=0 THEN PRINT "FIELD"L+1;F$(0,L)
3170 NEXT L
3180 INPUT "WHICH FIELD NUMBER";K: K=K-1: CLS
3190 IF K>B THEN 3140
3200 IF C(K)<>0 THEN 3140
3210 PRINT @64,"1 = ALL RECORDS"
3220 PRINT "0 = SELECTED RECORDS"
3230 INPUT "CHOOSE 1 OR 0";M: CLS
3240 IF M<>0 AND M<>1 THEN 3210
3250 AT=0: AN=0
3260 IF M=0 THEN 3320
3270 FOR L=1 TO R
3280 AT=AT+VAL(F$(L,K)): PRINT @239,AT
3290 IF VAL(F$(L,K))<>0 THEN AN=AN+1: PRINT @271,AN
3300 NEXT L
3310 GOTO 3330
3320 GOSUB 3470
3330 FOR L=1 TO 1000: NEXT L
3340 CLS: PRINT @64,"TOTAL OF "F$(0,K)
3350 IF M=0 THEN PRINT "FOR THE SELECTOR(S) GIVEN"
3360 PRINT "IS"AT: IF AT=0 THEN 3420
3370 IF M=1 THEN 3400
3380 PRINT "ON THE"AN"RELEVANT ENTRIES"
3390 PRINT "WITH AVERAGE ENTRY OF"AT/AN
3400 PRINT "AND AVERAGE OVER ALL"R
3410 PRINT "RECORDS OF"AT/R
3420 PRINT @448,"PRESS 'ENTER' WHEN READY...";
3430 INPUT A$: GOTO 1210
3440 REM >>>>>>>>>> selective print
3450 PRINT "YOU CAN HAVE A PRINTED TABLE OF ALL RECORDS WHICH COMPLY W
ITH   YOUR SPECIFICATIONS.": PRINT
3460 GOSUB 3470: PRINT #-2: GOTO 2100
3470 REM >>>>> selecting records
3480 PRINT "WORD KEYS CAN BE THEMSELVES OR  LEFT-END OR RIGHT-END."
3490 PRINT "ANY KEY CAN BE = OR < OR >.": PRINT
3500 PRINT "(INPUT 0 AT ANY STAGE TO ABORT  BACK TO THE MAIN MENU.)":
PRINT
3510 INPUT "DO YOU WISH TO SPECIFY          1 OR 2 OR 3 SELECTOR KEY(S
)";H: CLS
3520 IF H<0 OR H>3 THEN 3510
3530 IF H=0 THEN 1210
3540 FOR L=1 TO H
3550 FOR LL=0 TO B: PRINT LL+1;F$(0,LL): NEXT LL
3560 INPUT "WHICH FIELD NUMBER";Y(L): Y(L)=Y(L)-1 :CLS
3570 IF Y(L)>B THEN 3560
3580 IF Y(L)=-1 THEN 1210
3590 PRINT F$(0,Y(L))
3600 INPUT "INPUT THE SELECTOR KEY WORD OR  KEY VALUE";G$(L): CLS
3610 IF G$(L)="" THEN 3600
3620 IF C(Y(L))=0 THEN T(L)=1: GOTO 3700
3630 PRINT "WHICH CRITERION -"
3640 PRINT "1 = ONLY "G$(L)" AND ALL "G$(L)
```

```
3650 PRINT "2 = "G$(L)" AT LEFT END"
3660 PRINT "3 = "G$(L)" AT RIGHT END"
3670 INPUT T(L): PRINT
3680 IF T(L)<0 OR T(L)>3 THEN 3630
3690 IF T(L)=0 THEN 1210
3700 PRINT "INPUT THE REQUIRED SELECTOR -"
3710 PRINT "1 = "G$(L)" ONLY"
3720 PRINT "2 = UP TO "G$(L)
3730 PRINT "3 = AFTER/GREATER THAN "G$(L)
3740 INPUT V(L): CLS
3750 IF V(L)<0 OR V(L)>3 THEN 3700
3760 IF V(L)=0 THEN 1210
3770 NEXT L
3780 CLS
3790 FOR L=1 TO R
3800 PRINT L: P=1
3810 FOR LL=1 TO H
3820 IF P=0 THEN 4080
3830 ON T(LL) GOTO 3840,3850,3860
3840 ON V(LL) GOTO 3870,3890,3920
3850 ON V(LL) GOTO 3950,3970,3990
3860 ON V(LL) GOTO 4010,4030,4050
3870 IF F$(L,Y(LL))<>G$(LL) THEN P=0
3880 GOTO 4060
3890 IF C(Y(LL))=0 AND VAL(F$(L,Y(LL)))>VAL(G$(LL)) THEN P=0
3900 IF C(Y(LL))=1 AND F$(L,Y(LL))>G$(LL) THEN P=0
3910 GOTO 4060
3920 IF C(Y(LL))=0 AND VAL(F$(L,Y(LL)))<VAL(G$(LL)) THEN P=0
3930 IF C(Y(LL))=1 AND F$(L,Y(LL))<G$(LL) THEN P=0
3940 GOTO 4060
3950 IF LEFT$(F$(L,Y(LL)),LEN(G$(LL)))<>G$(LL) THEN P=0
3960 GOTO 4060
3970 IF LEFT$(F$(L,Y(LL)),LEN(G$(LL)))>G$(LL) THEN P=0
3980 GOTO 4060
3990 IF LEFT$(F$(L,Y(LL)),LEN(G$(LL)))<G$(LL) THEN P=0
4000 GOTO 4060
4010 IF RIGHT$(F$(L,Y(LL)),LEN(G$(LL)))<>G$(LL) THEN P=0
4020 GOTO 4060
4030 IF RIGHT$(F$(L,Y(LL)),LEN(G$(LL)))>G$(LL) THEN P=0
4040 GOTO 4060
4050 IF RIGHT$(F$(L,Y(LL)),LEN(G$(LL)))<G$(LL) THEN P=0
4060 NEXT LL
4070 IF P=1 THEN GOSUB 3080: IF J=3 THEN GOSUB 2500 ELSE AT=AT+VAL(F$(
L,K)): AN=AN+1
4080 NEXT L
4090 RETURN
4100 REM >>>>>>>>>> change main title
4110 PRINT "INPUT THE NEW TITLE"
4120 IF T$<>"" THEN PRINT "IN PLACE OF": PRINT T$
4130 PRINT: LINE INPUT T$: CLS
4140 PRINT @96,T$: PRINT
4150 INPUT "TYPE 'OK' OR 'NO'...";A$: CLS
4160 IF A$="OK" THEN 1210 ELSE 4110
4170 REM >>>>>>>>>> finished
4180 PRINT @192,"ARE YOU SURE YOU HAVE FINISHED? IF SO TYPE 'END', OR
ELSE PRESS 'ENTER' BY ITSELF."
4190 INPUT D$: CLS
4200 IF D$<>"END" THEN 1210
4210 PRINT @236,"finished": PRINT @268,"datasort"
4220 REM M.J. HOSKEN
```

VARIABLES used in DATASORT program – disk version

A	MAXIMUM NUMBER OF RECORDS = LENGTH OF ARRAY
A$	RE-USEABLE RESPONSE
AN	NUMBER OF NON-ZERO VALUES BEING TOTALLED AND AVERAGED
AT	TOTAL OF ALL VALUES DURING ARITHMETIC OPTION
B	NUMBER OF FIELDS PER RECORD MINUS 1 = WIDTH OF ARRAY
C(B)	FLAG: 0=NUMERICAL, 1=STRING
CR	FLAG: 1=AMEND RECORD WITH IDENTICAL FIRST FIELD
D$	RE-USEABLE RESPONSE
E	-
F$(A,B)	MAIN DATA ARRAY, A LONG BY B WIDE
F$(O,B)	FIELD TITLES - IF ANY
G$(3)	SELECTOR FOR PRINT
H	FLAG: 0=SOPHISTICATED SORT, 1=FAST SORT
H	NUMBER OF SELECTOR KEYS
H$	NAME OF DATA FILE (TEMPORARY)
H$(B)	SPARE ARRAY FOR DOING SERIES OF SWAPS
J	MAIN MENU SELECTOR
K	FIELD SELECTOR FOR SORTING
K	PRINT-TAB VALUE
KK	FLAG: 1=RECORDS HAVE BEEN COMBINED
L	LOOP CONTROL
LL	INNER LOOP CONTROL
LS	TERTIARY LOOP CONTROL
M	CONTROL ONE SORTED LIST DURING FILE COMBINING
M	FLAG: 0=INCREASING SORT, 1=DECREASING SORT
M	FLAG: 0=SELECTED RECORDS SUMMED, 1=ALL RECORDS SUMMED
M	PRINT MENU SELECTOR
M	RECORD SELECTOR FOR ALTERATIONS
M$	DATA TO BE FOUND BY SEARCH
N	CONTROL ONE SORTED LIST DURING FILE COMBINING
N	FIELD TO BE EXAMINED IN SEARCHING
P$	RECORD SELECTOR TO INSPECT/ALTER/DELETE
Q	FLAG: 1=SUCCESSFUL SEARCH
Q	FLAG: 1=COMBINE TWO DATA FILES
R	NUMBER OF RECORDS IN MEMORY
R(B)	GAPS TO LEFT OF FORMATTED FIELDS WHEN PRINTING
RP	NUMBER OF RECORDS PER PAGE
S	FLAG: 0=WITHOUT GAPS, 1=GAPS INCLUDED IN PRINT FORMAT
S	NUMBER OF RECORD TO BE PRINTED TO DISK FILE
SL(B)	STRING LENGTHS OF FIELDS
T$	PRINTABLE TITLE FOR WHOLE SYSTEM
T(3)	STRING-END SELECTOR ONLY/LEFT/RIGHT
U	FLAG: 0=FIELDS CAN GROW, 1=PRESET FIELD WIDTHS
V(3)	VALUE SELECTOR = < >
W	TOTAL WIDTH OF PRINT LINE OF LONGEST RECORD
X	CALCULATOR FOR FIXING POSITION OF INDICATOR CURSOR
Y	NUMBER OF RECORDS INPUT FROM DISK
Y(3)	FIELDS INVOLVED IN SELECTION
Z	FIELD AS CRITERION FOR COMBINING FILES
Z	STRING LENGTH DURING INPUT FROM DISK

Mountain building

Anthony Daniel *scales the hills with the help of contours*

IF you have booked for an unknown holiday destination, or are thinking of moving house, or you study geography, then have a look at this program which may be of use to you. By using a map with contours and a ruler, you can measure off the distance of the contours from the left hand edge of the ruler and reconstruct the shape of the hills or mountains you are viewing.

The program works to scale, but this does mean that the vertical height tends to be rather low and insignificant unless you are reconstructing the Alps or the Himalayas. To get round this problem it is possible to distort the vertical scale to exaggerate the hill size.

You can naturally print out any mountain drawn on the high resolution screen using a standard screen dump, but I have incorporated two ways in which a quicker print can be obtained using a Tandy CGP115 or similar printer. The first merely draws a line of the hill but the second way, obtained by pressing 'X' at the beginning of the program, shades the hill rather neatly as in the example, and also colours it on the screen.

I have set the default height between contour lines at 100 metres. This is probably more than on most maps you wish to measure but it makes testing the program when setting it up easier. If the total available height (as given on the screen) is inappropriate to the hills you are measuring you must alter the survey area. If you place yourself a long way from the hill it will appear much smaller. This is to enable you to see if a distant hill can be seen over the top of a nearer smaller hill. When reconstructing only one hill use the default distance of 1 km. In this section I have not taken into account the curvature of the earth but it is not difficult to do so if you are mathematically minded.

Once you have set up the initial criteria based on the map you are using read off the distance of the contour line from the left hand edge of the survey area (not the distance between contour lines) and enter the figure on the computer followed by and indication of the direction of the slope to follow, as instructed in the program.

Immediate corrections may be made. If you enter the slope wrongly simply re-enter it. If you put in an incorrect figure just backspace. However, to correct an error overlooked earlier on in the sequence end your reconstruction by pressing 'C', not 'F'. This takes you through a routine allowing correction of each individual entry as required.

Once the mountain is drawn press the space bar to discover the range of options available. As already mentioned you may reconstruct a second mountain or alter the vertical scale. If you wish to restore the original scale you must of course clear the screen first.

A program that changes scale in this way could be used to build up patterns and there are other applications that can be derived from it which have nothing to do with mountains, so I hope it will be of general interest and enjoyment.

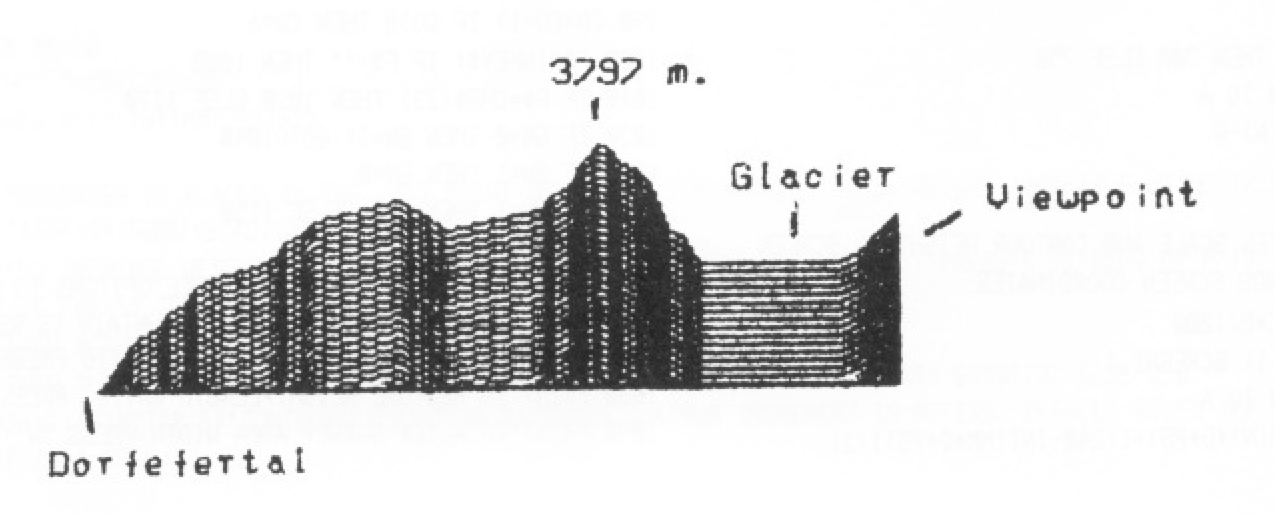
```
5 ' 'MOUNT' - CONTOUR RECONSTRUCTION PROGRAMME BY ANTHONY DANIEL - MARCH 1987
10 PRINT#-2,CHR$(18): PRINT#-2,"A": PRINT#-2,CHR$(18): 'PRINTER CODES FOR TANDY CGP115 PRINTER.
   DELETE IF PRINTER NOT IN USE
20 PCLEAR8
30 DIMB$(50),B(50),C(50),X(50),Y(50)
40 F=1: CO=6
50 PMODE3,1: SCREEN1,1: PCLS
60 CLS:PRINT:PRINT:PRINT"FOR SHADED PRINTINGS OF MOUNTAINSHAPES PRESS 'X' ELSE PRESS ANY OTHER KEY"
70 X$=INKEY$: IF X$="" THEN 70
80 IF X$="X" THEN EX=0 ELSE EX=1
90 CLS:PRINT:PRINT:PRINTTAB(5)"SCALE":PRINTTAB(5)"ENTER NO. OF MM. PER KM. ": PRINTTAB(5)"DEFAULT=20MM PER KM": INPUTSC
100 IF SC=0 THEN SC=20
110 GOSUB120: GOSUB160: GOTO280
120 PRINTTAB(5)"VIEWPOINT":PRINTTAB(5)"FOR COMPARING SEVERAL HILLSDISTANCE OF OBSERVER MUST BE PUTIN
   IN KILOMETRES":PRINTTAB(5)"DEFAU LT IS 1 KM (MIN. VIEW-ING DISTANCE)"
130 INPUT VP: IF VP=0 THEN VP=1
140 L(L)=VP
150 RETURN
160 CLS
170 PRINT:PRINT:PRINTTAB(5)"SURVEY AREA WIDTH IN MM. ASCOVERED BY RULER ON MAP":PRINTTAB(5)"DEFAULT GIVES 60MM."
180 INPUT MM
190 IF MM=0 THEN MM=60
200 IF MM<255 THEN D=INT(255/MM)
210 IF MM>254 THEN D=1/(INT(MM/255)+1)
220 GOSUB1760
230 PRINTTAB(5)"OBSERVER IS PLACED IN THE  MID POINT OF THIS WIDTH.":PRINTTAB(5)"HEIGHT AVAILABLE USING THISSCALE
   AND AREA = ";INT(19 0000/(SC*D));" METRES"
240 PRINTTAB(5)"TO INCREASE HEIGHT INCREASESURVEY AREA OR VICE VERSA": PRINTTAB(5)"PRESS R TO REDO OR P TO    PROCEED"
250 PM$=INKEY$: IF PM$="" THEN 250
260 IF PM$="P" THEN RETURN
270 IF PM$="R" THEN CLS: PRINTTAB(5)"UNREVISED SURVEY AREA IS    ";MM;"MM.": PRINT: GOTO170: ELSE 250
280 CLS:PRINT:PRINT:PRINTTAB(5)"CONTOURS":PRINTTAB(5)"ENTER CONTOUR INCREMENT IN METERS. DEFAULT GIVES 100 M."
290 INPUT CN: IF CN=0 THEN CN=100
```

```
300 PRINTTAB(5)"NOTE:- IF SEVERAL HILLS ARETO BE COMPARED THE STARTING     HEIGHT USED SHOULD BE THE
    OBSERVER'S HEIGHT. IF THE SHAPE OF ONLY ONE HILL IS REQUIRED    THIS IS IRRELEVENT"
310 CLS:PRINT:PRINTTAB(5)"PLACE A MM RULER ACCROSS   THE LINE OF CONTOURS TO BE SUR- VEYED. ENTER THE DISTANCE OF
    EACH CONTOUR FRO M THE left hand edge of the ruler."
320 PRINTTAB(5)"SEPARATE EACH DISTANCE BY  THE SLOPE DIRECTION. ';'= UPWARD'-'= DOWNWARD AND 'ə'= LEVEL":PRINT"
    (SIGNS ON SCREEN ARE +  , - AND =)"
330 PRINTTAB(5)"TO END THE SEQUENCE PRESS  'F'. PRIOR TO PRESSING F BEGIN  AND END WITH A NUMBER.
    SEQUENCESMUST BE AT LEAST 4 NUMBERS  LONG.";
340 PRINTTAB(5)"PRESS SPACEBAR TO CONTINUE THROUGHOUT PROGRAMME";
350 IF INKEY$=CHR$(32) THEN 360 ELSE 350
360 A$=""
370 CLS: M=33
380 PRINTəM,A$;: A$="": GOSUB1720
390 A$=INKEY$: IF A$="" THEN 390
400 IF A$="C" GOSUB1420: GOTO1510
410 IF A$="F" AND PV=1 THEN 390
420 IF A$="F" THEN GOSUB1420: GOTO630
430 IF A$=CHR$(8) AND PEEK(1024+M)<>107 AND PEEK(1024+M)<>109 AND PEEK(1024+M)<>125 THEN PRINTəM,CHR$(143);:
    M=M-1: GOTO440 ELSE 460
 40 IF LEN(B$(A))       >1 THEN B$(A)=LEFT$(B$(A),LEN(B$(A))-1) ELSE B$(A)=""

450 GOTO390
460 IF ASC(A$)>47 AND ASC(A$)<58 THEN 530
470 IF A$=";" THEN A$="+"
480 IF A$="ə" THEN A$="="
490 IF E$="+" OR E$="-" OR E$="=" THEN PV=1
500 IF A$="+" OR A$="-" OR A$="=" THEN PW=1:
    GOTO510: ELSE390
510 IF PV=1 AND PW=1 THEN E$=A$: GOTO380
520 E$=A$: M=M+1: GOTO380
530 IF PW=1 THEN GOSUB550
540 E$=A$:B$(A)=B$(A)+A$: M=M+1: GOTO380
550 IF E$="+" THEN GOSUB1420 ELSE 570
560 C(A+1)=C+1:C=C(A+1):B$(A)="":GOSUB1450:GOTO610
570 IF E$="-"  THEN GOSUB1420 ELSE 590
580 C(A+1)=C-1:C=C(A+1):B$(A)="":GOSUB1450:GOTO610
590 IF E$="="  THEN GOSUB1420 ELSE 620
600 C(A+1)=C:C=C(A+1):B$(A)="": GOSUB1450:GOTO610
610 PV=0:PW=0
620 RETURN
630 ' FINDS HIGHEST AND LOWEST POINTS
640 FOR N=0 TO A
650 IF N=0 THEN Z=C(N): W=C(N): GOTO680
660 IF C(N)>Z THEN Z=C(N): J=N
670 IF C(N)<W THEN W=C(N)
680 NEXT
690 IF W<0 THEN 700 ELSE 750
700 FOR N=0 TO A
710 C(N)=C(N)-W
720 NEXT
730 Z=Z-W
740 ' RELATES SCALE AND CONTOUR HEIGHT TO SCREEN
    AND FINDS SCREEN COORDINATES
750 G=CN*SC*D/1000
760 PMODE3,1: SCREEN1,1
770 FOR N=0 TO A
780 X(N)=(B(N)*D*PS)+((240-INT(MM*D*PS))/2)
790 Y(N)=190-INT(PS*C(N)*G*F)
800 GOSUB1460
810 NEXT
820 FOR N=0 TO A
830 PSET(X(N),Y(N))
840 IF N=0 THEN 850 ELSE 880
850 IF Y(A)<Y(0) THEN LINE(X(A),Y(A))-(X(A),Y(0)),PSET:
    LINE(X(A),Y(0))-(X(0),Y(0)),PSET: GOTO890
860 IF Y(A)>Y(0) THEN LINE(X(A),Y(A))-(X(0),Y(A)),PSET:
    LINE(X(0),Y(A))-(X(0),Y(0)),PSET:GOTO890: ELSE LINE
    (X(A),Y(A))-(X(0),Y(0)),PS ET
870 GOTO890
880 LINE(X(N-1),Y(N-1))-(X(N),Y(N)),PSET
890 NEXT
900 IF EX=1 THEN 1000
910 ' PAINTS MOUNTAIN ON SCREEN IF REQUIRED. PRESS 'X'
    FOR THIS AT BEGINNING OF PROGRAMME
920 FOR N=1 TO A-1
930 IF Y(N)>Y(N-1) THEN Y=Y(N)-2
940 IF Y(N)=Y(N-1) THEN NEXT
950 IF Y(N)<Y(N-1) THEN Y=Y(N)+2
960 X=X(N)
970 IF PPOINT(X,Y)=5 THEN PAINT(X,Y),CO,8: GOTO990 ELSE 980
980 NEXT
990 CO=CO+1: IF CO>8 THEN CO=6
1000 F$=INKEY$: IF F$="" THEN 1000
1010 IF F$=CHR$(32) THEN 1020 ELSE 1170
1020 IF GN=0 THEN GN=1: GOTO1040
1030 IF GN=1 THEN GN=0
1040 IF GN=1 THEN CLS: ELSE 1150
1050 IF L>0 THEN 1100
1060 'important note - THE IST THREE OPTIONS TO FOLLOW ARE
     ONLY AVAILABLE WHEN ONLY ONE MOUNTAIN IS RECONSTRUCTED
1070 PRINT"TO INCREASE HEIGHT/LENGTH RATIO PRESS 'H'"
1080 PRINT"TO RESTORE HEIGHT/LENGTH RATIO  PRESS 'R'"
1090 PRINT"TO ALTER SURVEY AREA WIDTH PRESS 'W'"
```

```
1100 PRINT"TO RECONSTRUCT ANOTHER MOUNTAIN PRESS 'M'"
1110 PRINT"TO REVIEW PICTURE PRESS SPACEBAR"
1120 PRINT"TO PRINT OUT PRESS 'P'"
1130 PRINT"TO CLEAR SCREEN PRESS 'C'"
1140 GOTO1000
1150 PMODE3,1: SCREEN1,1
1160 GOTO1000
1170 IF L>0 THEN 1240
1180 ' 'F' IS THE VERTICAL SCALING FACTOR
1190 IF F$="H" THEN F=F+.2: GOTO760
1200 IF F$="W" THEN 1210 ELSE 1230
1210 GOSUB160
1220 GOTO750
1230 IF F$="R" THEN F=1: GOTO760
1240 IF F$="C" THEN PMODE3,1: SCREEN1,1: PCLS: GOTO1000
1250 IF F$="M" THEN 1260 ELSE 1310
1260 FOR N=0 TO A
1270 B$(N)="": B(N)=0: C(N)=0: X(N)=0: Y(N)=0
1280 NEXT
1290 L=L+1: CLS: GOSUB120
1300 GOTO280
1310 IF F$="P" THEN 1320 ELSE 1410
1320 E=0
1330 FOR N=0 TO A
1340 IF EX=0 THEN GOSUB1790 ELSE EA=Y(N)
1350 IF N=0 THEN PRINT#-2,"M";X(N);",";-EA: ELSE
     PRINT#-2,"D";X(N);",";-EA
1360 NEXT
1370 IF Y(A)>Y(0) THEN PRINT#-2,"D";X(0);",";-Y(A):
     PRINT#-2,"D";X(0);",";-Y(0)
1380 IF Y(A)<Y(0) THEN PRINT#-2,"D";X(A);",";-Y(0):
     PRINT#-2,"D";X(0);",";-Y(0)
1390 IF Y(A)=Y(0) THEN PRINT#-2,"D";X(0);",";-Y(0)
1400 IF ZZ=1 AND L=0 THEN E=E+2: ZZ=0: GOTO1330 ELSE 1000
1410 GOTO1000
1420 ' THIS IS THE HORIZONTAL MEASUREMENT
1430 B(A)=VAL(B$(A))
1440 RETURN
1450 A=A+1: RETURN
1460 IF X(N)<0 THEN X(N)=0
1470 IF X(N)>255 THEN X(N)=255
1480 IF Y(N)<0 THEN Y(N)=0
1490 IF Y(N)>191 THEN Y(N)=191
1500 RETURN
1510 FOR U=416 TO 510
1520 PRINT@U,CHR$(143);
1530 NEXT
1540 ' CORRECTION ROUTINE FOR INCORRECT ENTRIES. I HAVN'T
     INCLUDED FACILITY FOR SLOPE CORRECTION. YOU HAD BETTER
     START AGAIN IF YOU H AVE GOT THAT WRONG AS WELL!
1550 PRINT@320,"TO CHECK OR CORRECT ENTRIES"
1560 PRINT@483,"PRESS C FOR CORRECT, F FOR FAULTY";
1570 H$="EYE LEVEL"
1580 FOR N=0 TO A
1590 IF N=0 THEN 1640
1600 IF C(N)>C(N-1) THEN H$="RISING"
1610 IF C(N)<C(N-1) THEN H$="FALLING"
1620 IF C(N)=C(0) THEN H$="EYE LEVEL"
1630 IF C(N)=C(N-1) THEN H$="LEVEL"
1640 PRINT@352,"READING";B(N),"HEIGHT";H$
1650 M$=INKEY$: IF M$="" THEN 1650
1660 IF M$="C" THEN 1700
1670 IF M$="F" THEN 1680 ELSE 1650
1680 INPUT"READING";B(N)
1690 PRINT@383,"";
1700 NEXT
1710 GOTO630
1720 ED=INT(190000/(SC*D))
1730 PRINT@416,"HEIGHT AVAILABLE ";ED
1740 PRINT@448,"SURVEY AREA",MM
1750 PRINT@480,"HEIGHT ";C(A)*CN;
1760 IF L=0 THEN PS=1: RETURN
1770 PS=L(L-1)/L(L): RETURN
1780 ' THESE LINES ARE CONCERNED WITH SHADED PRINTINGS
     ON THE CGP115 PRINTER
1790 EA=Y(N)+E:
1800 IF Y(A)>Y(0) THEN EB=Y(A) ELSE EB=Y(0)
1810 IF EA>EB THEN EA=EB
1820 IF EA<EB THEN ZZ=1
1830 RETURN
```

The Grossglockner Mountain in Austria



Dragon Answers

If you've got a technical question write to Brian Cadge. Please do not send a SAE as Brian cannot guarantee to answer individual inquiries.

Save/Load

I write on the subject of save/load sequences for the Tandy/Dragon machines. I am converting the following code and need to know the same for Dragon Basic.

```
OPEN "D", #1,FILE$+"/GPH",33
FOR RUM=1 TO RNUM
PRINT #-1,TEST(RUM):
PUT #-1,RUM
NEXT RUM
PRINT #-1,NAME$:PUT #-1
PRINT #-1,BOT$:PUT #-1
PRINT #-1,SIDE$:PUT #-1
CLOSE #-1

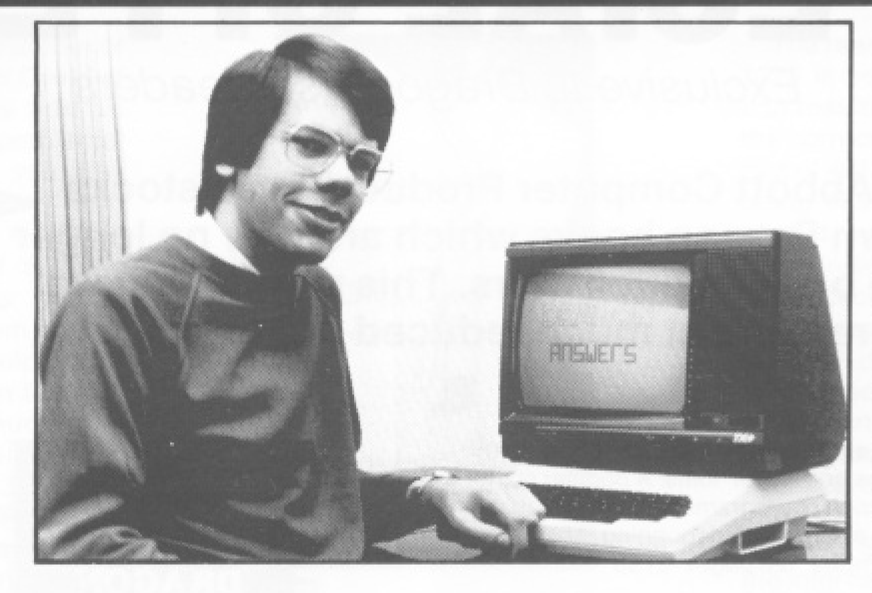
OPEN "D", #1,FILE$+"/GPH",33
FOR RUM=1 TO LOF(1)-3
GET #1:INPUT #1,TEST(RUM)
NEXT RUM
GET #1:INPUT #1,NAME$
GET #1:INPUT #1,BOT$
GET #1:INPUT #1,SIDE$
```

Maarten Van Wamelen
3 Lynmetestraat
Belgium

THESE two routines are simply saving an array of numbers, followed by three strings to a serial disc file on the CoCo. You can use the following DragonDOS routines in their place:

```
CREATE FI$+".GPH"
FWRITE FI$+".GPH";RNUM
FOR RUM=1 TO RNUM
FWRITE FI$+".GPH";TEST(RUM)
NEXT RUM
FWRITE FI$+".GPH";NAME$
FWRITE FI$+".GPH";BOT$
FWRITE FI$+".GPH";SIDE$
CLOSE

FREAD FI$+".GPH";RNUM
FOR RUM=1 TO RNUM
FREAD FI$+".GPH";TEST(RUM)
NEXT I
FLREAD FI$+".GPH";NAME$
FLREAD FI$+".GPH";BOT$
FLREAD FI$+".GPH";SIDE$
CLOSE
```



Empty printer

I HAVE recently been offered the use of a redundant Amstrad PCW 8512 printer. Would it be possible to adapt this to work with my Dragon 32? If so, what type of lead and power source do I need?

Roy Dean
53 Lulworth Road
Eccles
Manchester

THE problem with the PCW printer is that most of the electronics and the power supply are not in the printer itself but in the PCW computer. Therefore it would be very difficult to connect these printers to any other computer (and somewhat expensive if it's even possible). However, one of our readers may know otherwise...?

More windows

WITH regard to your *Windows* program published in the July 1985 issue of *Dragon User*, is it possible to modify this to use more windows (eg 5 to 9) as I find 5 is not enough?

Chris Steele
1 Spey Walk
Holytown
Lanarkshire
Scotland

IT'S certainly possible to add more windows, but you'll need to re-assemble the program to do it. First of all set up more data areas at the end of the code, each 7 bytes long (label them WIN6 to WIN10) and insert them just before the USRLOC label line.

Next you'll need to change all the error checks which assume only five windows, ie the lines at addresses 7CD6 and 7D49 in the listing. Finally, re-assemble the listing and save to tape.

Raves from the grave

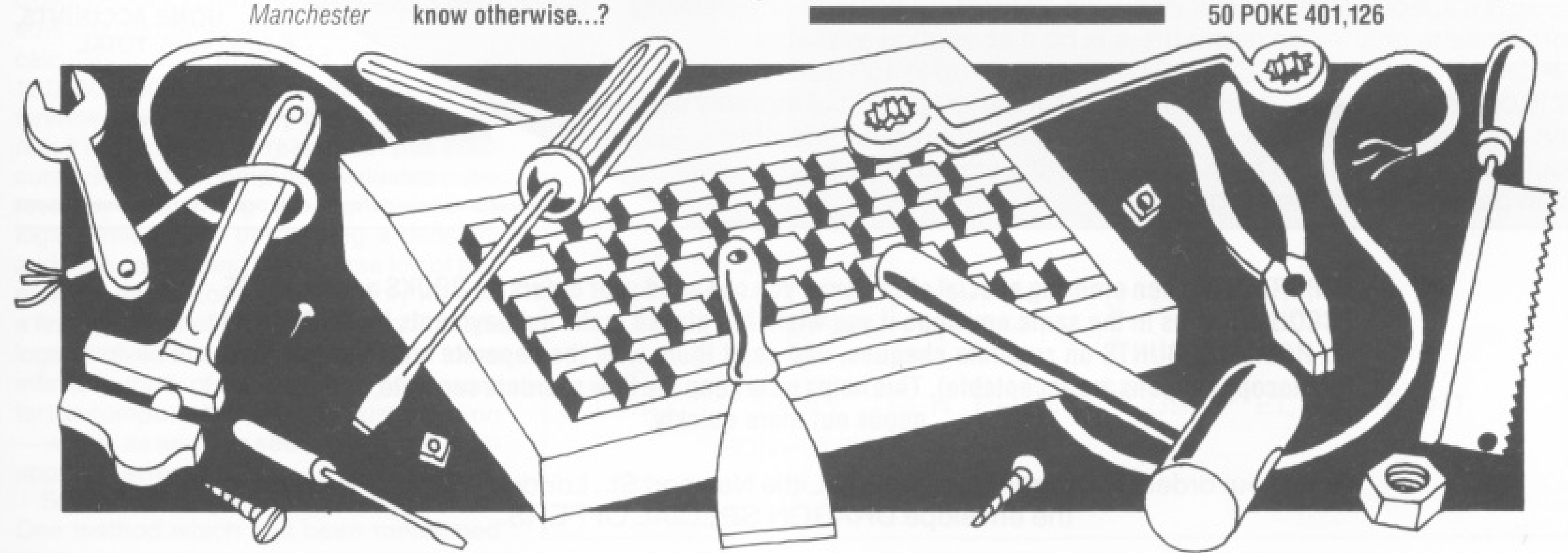
DUE to the postal problems in the London area we're trather short on readers enquiries this month. Therefore, I am taking the opportunity to reprint a couple of small routines which I'm regularly asked about.

This first routine originally appeared in the March 1984 issue of *Dragon User*. If placed at the beginning of your program it will disable both the LIST and LLIST commands so that, used in conjunction with an AUTORUN routine, your code is protected from prying eyes.

```
10 CLEAR 200,32620
20 TS=PEEK(291)*256+PEEK(292)
30 FOR I=0 TO 117
40 POKE 32650+I,PEEK(I+TS)
50 NEXT I
60 'DISABLE LIST COMMAND
70 POKE 32650+42,134
80 POKE 32650+43,22
90 'DISABLE LLIST COMMAND
100 POKE 32650+58,134
110 POKE 32650+59,22
120 FOR I=0 TO 6:READ A$:POKE
   32621+I,VAL("&H"+A$):NEXT
130 DATA 8E,7F,8A,BF,01,23,39
140 EXEC 32621
150 'REST OF PROGRAM ....
```

THIS second routine originally appeared in the December 1983 issue. It provides an ON ERROR RUN facility so that if an error occurs the program simply re-runs itself. This is useful for programs that will be run by non-computer users to whom a '?/0 ERROR' could be quite alarming! It can be disabled by POKE 401,57 and re-enabled by POKE 401,126.

```
10 CLEAR 200,32749
20 FOR I=1 TO 10:READ A$:
   POKE 32749+I,VAL('&H'+A$):
   NEXT
30 DATA 32,62,BD,84,IF,BD,83,ED,
   7E,84,9F
40 POKE 402,127:POKE 403,238
50 POKE 401,126
```

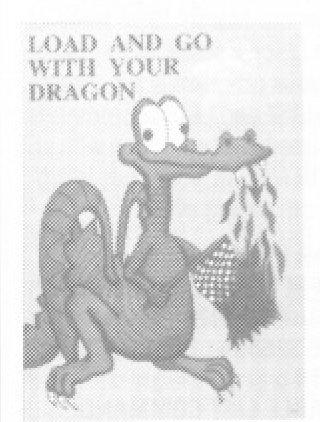


SPECIAL OFFER!

Exclusive to Dragon User readers

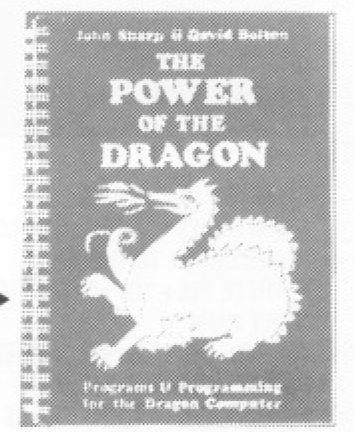
FREE GAME with first 450 orders

Dragon User and Abbott Computer Products have stocks of three well known Dragon books which are now no longer available from the original publishers. This means that we can offer them to readers at much reduced prices while stocks last.



◀ **Load and Go with your Dragon** by John Phipps and Trevor Toms. A popular starter's collection of programs with educational hints, a section on debugging, games, and a Wumpus. 128 pages. £1.95

The Power of the Dragon by John Sharp and David Bolton. A guide to programming the Dragon, undertaken through a collection of 30 entertaining programs from games to VAT calculation. 175 pages, £2.75 ▶



The first 450 orders will also receive a FREE GAME — Bonka is a ladders-and-platforms game for the Dragon 32. It's not one of the great DRAGON classics. It's not even new. But it's a great way to kill off those idle half hours when you feel like a break from reading or programming . . . and it's free.

These books have been travelling around looking for Dragon owners for some time, so although they are in perfect nick inside, most of the covers have a tiny dent, or sticker mark, or other small token of life's experience. These, like the low prices, are all part of the natural beauty of cut-price books, and do not affect their performance.

Dragon Magic — Your first programming book by Richard Wadman. This is a programming primer, approved as a Dragon teaching book by Dragon Data, ideal for youngsters (or not so youngsters) starting out. With line and colour illustrations. 56 pages, £1.50 ▶

Please send me the following DRAGON books:

..... **LOAD AND GO**, £1.95 each .....
..... **THE POWER OF THE DRAGON**, £2.75 each .....
..... **DRAGON MAGIC**, £1.50 each .....
TOTAL .....

NAME .... **ADDRESS** ....
.... **POSTCODE** ....

Please make out cheques and postal orders to Scot Press Ltd. (Dragon books)

Bank account empty? Don't know where the money goes? Can't afford an accountant? Or even a lock-up wallet?

You should have DRAGON HOME ACCOUNTS for the Dragon 32.

- A 4pp booklet describes the options for storing your monthly household transactions on tape, making forward estimates of expenditure and reviewing your budget to date or month by month.
- The accompanying program was originally marketed by Gemini Marketing at £19.19. DRAGON USER's special offer price is £6.50, INCLUSIVE of postage and packing. The program comes on cassette only — we regret there is no disc version available.
- DRAGON HOME ACCOUNTS can handle up to 150 transactions a month of up to £10,000 each. The program can display a bar graph plot of monthly expenditure on any particular item, and store fixed values for regular monthly outgoings. Information from earlier months stored on separate tapes can be loaded into the program.

Please send me .....copies of **DRAGON HOME ACCOUNTS**, at £6.60 each. **TOTAL** .....

NAME ....
ADDRESS ....
....

Please make out cheques or postal orders to Scot Press Ltd. (Home Accounts)

IMPORTANT: When ordering special offer items, you can send your orders for BOOKS and for HOME ACCOUNTS to us in the same envelope if you wish, but please make out payments for BOOKS and for HOME ACCOUNTS on separate cheques, and send them with the separate coupons provided (photocopy coupons are acceptable). This helps us to keep the flow of orders separate, and send the goods out more quickly.

Send your orders to Dragon User at 12/13 Little Newport St., London WC2H 7PP, marking the envelope DRAGON SPECIAL OFFERS.

Winners and Losers

Every month, Gordon Lee will look at some prize programming points from a previous month's competition

A request for assistance from a number of readers, in particular from Keith David of Crawley, who is having difficulty with his cube roots! Well, we all have our problems!

"Please," he writes, "can you say something about testing for cubes?"

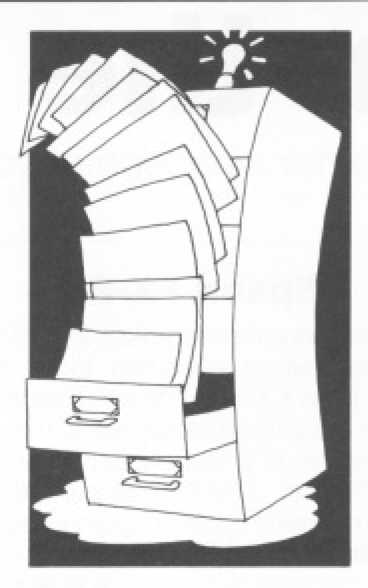
Like many competitors, he had discovered that determining if a given number is a perfect cube is not as straightforward as it might seem at first sight. The logical method would be to evaluate the cube root and then test this number to see if it was a whole number — perhaps using a simple routine such as: R=N↑(⅓): IF R=INT(R) THEN .... , where N is the number under test and R its cube root. (Note to non-mathematicians: Raising a number to a power which is a reciprocal is equivalent to finding the appropriate root of that number).

Unfortunately, with the Dragon, as with most micros, this doesn't always work. readers who are interested might like to try **listing 1**, which tests the cubes of the numbers from 1 to 999. The program itself is quite simple, and yet it just doesn't work! Starting with a number N, its cube is found by direct multiplications (C). The cube root (R) is then calculated using the line R=C↑(⅓). In an ideal world R should, not surprisingly, be equal to the original value N, and so line 40 should print out a complete list of the integers from 1 to 999. In fact, it succeeds in printing just 119 of these numbers — so 880 of the numbers tested have resulted in an error being made by the computer.

So what has gone wrong? Take the number 1728, the cube of 12. If we type in PRINT 1728↑(⅓), the result, 12 is printed onto the screen, so all would appear well. Now try IF 1728↑(⅓)=12 THEN PRINT"CORRECT". As this results in no message being printed, the computer clearly regards the two values as not being equal — which we know, mathematically, they are.

To get a clue as to what is happening type PRINT 1728↑(⅓)-12. Instead of the expected zero we get the result 7.4505806E-09. This strange number, and note the E-09 at the end, actually represents the value 0.0000000074505806. What has happened is that the computer, due to its method of calculation, has evaluated the result as 12.0000000074505806, and this minor difference from 12 exactly has affected the result of the test. The reason for this inaccuracy is that the computer evaluates cube roots by converting the number to its logarithmic value, performing a division, and then calculating the inverse log of this number. As the computer can only work to a finite number of decimal places, and the logs involved are irrational and exist to an infinite number of decimals, it is necessary for the computer to make an approximation — which, as we have seen, can result in an apparent error.

So how do we get around this problem? One method which has been mentioned



before in the magazine, is to convert the value into a string variable, and then back again into a numeric variable, before performing the test. In **listing 1** this would be effected by inserting line 35:

35 R$=STR$(R):R=VAL(R$)

Unfortunately, although this improves the performance of the program, it is still not infallible. The success rate rises to 557 out of 999, so clearly a better method of approach is required.

My solution is given as **listing 2**, which relies on the fact that a straight multiplication (as in line 1030) is exact, at least as far as a nine-digit product is concerned. The method that is used is quite simple:

i) Find the approximate root
ii) Reduce it to its integer
iii) Cube this result and compare with the number under test.

The reason for testing two values at line 1020 is because sometimes the calculation results in a value which is just under the correct answer. For example, the cube root of 4913, the cube of 17, is assessed as being 16.999999...so the extra step is included to cover this eventually. The actual test is indicated by setting a flag (FL) to 1 if an exact comparison is made.

With regards to the competition itself, the main difficulty, apart from the one just mentioned, was the length of time that some starting values needed to run before coming up with a perfect cube. In particular, a starting value of 19 will not terminate until a score of 125000 has been reached. Several entrants realised that as the interval between each successive pair of cubes gets progressivly larger, there comes a point when a method of speeding up the calculation is desirable. For example, a complete circuit of the board, scoring 210 points, can sometimes be added in one operation.

For an alternative method of solution please refer to my own solution on page 6 of August's issue.

Finally, a couple of corrections to July's *Winners and Losers*. The number of possible grids is 10^{36} — not 1036 as printed — no small difference!!! Also the dissection of the grid as given towards the end of the page should read:

"...in a 6x6 grid there are 14 six-digit numbers, 32 five-digit numbers, 54 four-digit numbers, 80 three-digit numbers, and 110 two-digit numbers. This makes a total of 290 numbers. Each of these can also be read in reverse, making 580 in all. Finally, there are 36 single-digit numbers, bringing the grand total up to 616."

Listing 1

```
10 FOR N=1 TO 999
20 C=N*N*N
30 R=C↑(1/3)
40 IF R=N THEN PRINT N,C
50 NEXT N
```

Listing 2

```
1000 INPUT V
1010 R=V↑(1/3)
1020 FL=0:FOR F=INT(R) TO INT(R+1)
1030 IF F*F*F=V THEN FL=1
1040 NEXT F
1050 IF FL=1 THEN PRINT "CUBE" ELSE PRINT
     "NON-CUBE"
```

Down in the dumps

***Dragon User** presents another selection of screen dumps from readers*

Epson FX-80

The following routine pokes in 219 bits of data and when executed looks at the screen in an X Y format to print out the contents of the screen. Attached to the end of the program is a sample listing numbers from 1000 on. The normal procedure for saving machine code programs should be used after proving the procedure works.

A.R.Butterfield

```
LISTING 1

10 CLEAR 200,&H7CFF

20 X=&H7E00

30 READ X$

40 IF X>32767 OR X<&H7CFF THEN STOP

50 IF X$="ZZ" THEN 80

60 POKE X,VAL("&H"+X$)

70 X=X+1:GOTO 30

80 Y=&H7F00

90 READ X$

100 IF X>32767 OR X<&H7CFF THEN STOP

110 IF X$="ZZ" THEN 140

120 POKE Y,VAL("&H"+X$)

130 Y=Y+1:GOTO 90

140 GOTO 1000

150 DATA34,36,BE,1D,E0,10,BE,7D,00,10,BF,7F,E3,BD,7F,00,C6,C0,10,BE

160 DATA7D,00,F7,7F,E1,A6,84,43,BD,7F,26,BD,80,0F,86,00,BD,80,0F,30

170 DATA88,E0,7A,7F,E1,F6,7F,E1,C1,00,26,E5,86,0D,BD,80,0F,10,8E,7D

180 DATA00,10,BF,7F,E3,BD,7F,4F,30,89,18,01,8C,1E,00,26,C0,35,36,39

190 DATAZZ,86,1B,BD,80,0F,86,41,BD,80,0F,86,07,BD,80,0F,86,1B,BD,80

200 DATA0F,86,4B,BD,80,0F,86,80,BD,80,0F,B6,01,BD,80,0F,39,12,12,12

210 DATA5F,46,56,56,46,56,56,46,56,56,46,56,E7,A4,7C,7F,E4,10,BE,7F

220 DATAE3,5F,46,56,56,46,56,56,46,56,56,46,56,F7,7F,E2,B6,7F,E2,39

230 DATA12,12,BD,7F,00,C6,C0,F7,7F,E1,A6,A4,12,BD,80,0F,86,00,BD,80

240 DATA 0F,31,A8,01,7A,7F,E1,F6,7F,E1,C1,00,26,E8,86,0D,BD,80,0F,39

250 DATA "ZZ"

1000 PMODE4,1:PCLS1:SCREEN1,0

1010 N=2:PI=4*ATN(1)

1020 Y0=80/N/N

1030 DRAW"BM,"+STR$(INT(186-Y0*2))

1040 FOR X=0 TO 255 STEP 2

1050 Y=186-Y0-Y0*COS(N*2*PI*X/255)

1060 LINE-(X,Y),PSET

1070 XT=X+2*Y0*N*PI*Y/255*SIN(N*2*PI*X/255)

1080 Y1=0:IF XT<0 THEN XT=0:YZ=Y+255*X/(2*N*PI*Y0*SIN(N*2*PI*X/255))

1090 IF XT>255 THEN XT=255:Y1=-(255-X)*255/(2*N*PI*Y0*SIN(N*2*PI*X/255))

2000 LINE-(XT,Y1),PRESET

2010 DRAW"BM"+STR$(X)+","+STR$(INT(Y))

2020 NEXT

2030 EXEC&H7E00
```

Tandy Color Graphic Printer

This program is a screen dump for the Tandy Color Graphic Printer, though it may work on other printer/plotters. It uses all four colours and dumps sideways on the paper, in order to give the largest size. Any colour that is not available on the printer is shown as white. The dump is not solid colour, but made up of a series of minute squares, so saving ink in the pens. If your machine can take it then it is advisable to use the speed poke (POKE 65495,0) (*Remember you have been warned against using the speed poke — Ed.*). Even so while the screen is being dumped, it is advisable to leave it and do something else. A whole screen can take around 40 minutes, depending on the number of colour changes.

J.F.Singleton

```
10 REM (C) J.F.SINGLETON
20 REM JOHN TO MY FRIENDS
30 CLS
40 INPUT "SWITCH PRINTER ON THEN ENTER";
A$
50 PRINT#-2,CHR$(18)
60 INPUT "WHAT PMODE IS THE PICTURE ON";
PM
70 INPUT "WHAT PAGE IS THE PICURE ON";PA
80 INPUT "WHAT COULOUR SET IS IT IN";CS
90 INPUT "TOP LEFT CO-ORDINATES OF AREA
TOBE DUMPED (X,Y)";TX,TY
100 INPUT "BOTTOM RIGHT CO-ORDINATES OF
   AREA TO BE DUMPED (X,Y)";BX,BY
110 IF PM<1 OR PM>4 THEN PRINT "INVALID
ENTRY, DO AGAIN":GOTO 190
120 IF PA<1 OR PA>8 THEN PRINT "INVALID
ENTRY, DO AGAIN":GOTO 190
130 IF TX<0 OR TX>256 THEN PRINT "INVALI
D TOP CO-ORDINATES, DO    AGAIN":GOTO 19
0
140 IF TY<0 OR TY>192 THEN PRINT "INVALI
D TOP CO-ORDINATES, DO    AGAIN":GOTO 19
0
150 IF BX<0 OR BX>256THEN PRINT "INVALID
 BOTTOM CO-ORDINATES, DO AGAIN":GOTO 190
160 IF BY<0 OR BY>192 THEN PRINT "INVALI
D BOTTOM CO-ORDINATES, DO AGAIN":GOTO 19
0
170 IF CS<0 OR CS>1 THEN PRINT "INVALID
COLOUR SET, DO AGAIN":GOTO 190
180 IF CS=1 AND PM=1 OR PM=3 THEN PRINT
"YOUR DUMP WILL BE ALL WHITE, USEA DIFFE
RENT COLOUR SET":GOTO 190
190 SOUND 50,5:FOR I=1 TO 1000:NEXT:RUN
200 PMODE PM,PA:SCREEN1,CS
210 HE=3*(BY-TY)
220 PRINT#-2,"R0,-";HE
230 PRINT#-2,"I"
240 FOR Y=TY TO BY
250 FOR X=TX TO BX
260 A=PPOINT (X,Y)
270 IF A=0 THEN PRINT#-2,"C0":GOTO 320
280 IF A=1 THEN PRINT#-2,"C2":GOTO 320
290 IF A=3 THEN PRINT#-2,"C1":GOTO 320
300 IF A=4 THEN PRINT #-2,"C3":GOTO320
310 GOTO 370
320 PRINT#-2,"J2,0,0,2,-2,0"
330 NEXT X
340 PRINT#-2,"H,R2,0"
350 PRINT#-2,"I"
360 NEXT Y:END
370 PRINT#-2,"R0,2"
380 GOTO 330
```

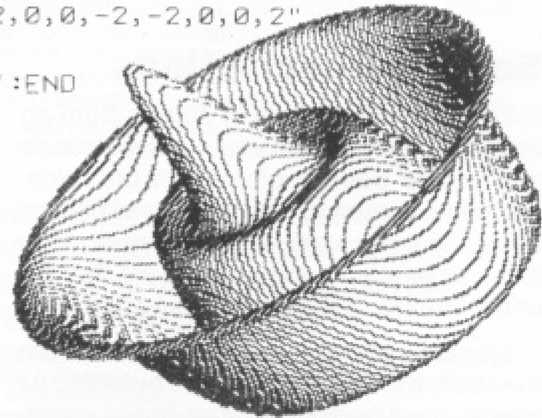
Tandy CGP-115

This is a two colour screen dump for the Tandy CGP-115 printer. In the first part the foreground oclour and colour set are defined by the user, in addition to this is user is able to choose the colour of the hardcopy.

The main part is from line 170 onwards. These lines set the printer to graphics mode and select the drawing colour. As the printer paper is not wide enough to allow a double size screen dump the normal way, line 250 changes the X co-ordinate so the picture is drawn vertically down the paper. The screen dump enclosed took just over 2 hours.

P. Marlow

```
10 '(C) 30/5/87 P. MARLOW
20 PMODE 4,1
30 CLS
40 PRINT TAB(1);"CGP-115 TWO COLOUR SCRE
EN DUMP"
50 PRINT TAB(1);STRING$(30,"=")
60 PRINT:PRINT
70 INPUT "WHICH FOREGROUND COLOUR (0/1/5
)";FC
80 IF FC<>0 AND FC<>1 AND FC<>5 THEN 30
90 INPUT "WHICH SCREEN (0/1)";SC
100 PRINT@256,"CHOOSE HARDCOPY COLOUR"
110 PRINT "0:- BLACK":PRINT"1:- BLUE":PR
INT"2:- GREEN":PRINT"3:- RED"
120 INPUT "ENTER CHOICE";HC
130 IF HC<0 OR HC>3 THEN 100
140 CLS:INPUT "LOAD PICTURE FROM TAPE (Y
/N)";A$
150 IF A$="Y" THEN PRINT "INSERT CASSETT
E, PRESS PLAY":INPUT "ENTER PICTURE TITL
E";TI$:SCREEN 1,SC:CLOADM TI$
160 SCREEN 1,SC
170 PRINT #-2,CHR$(17)
180 PRINT #-2,CHR$(18)
190 PRINT #-2,"1"
200 PRINT #-2,"C";HC
210 FOR Y=0 TO 191
220 FOR X=0 TO 255
230 P=PPOINT(X,Y)
240 IF P<>FC THEN 280
250 V=(X*-2)+510
260 PRINT #-2,"M";Y*2;",";-V
270 PRINT #-2,"J2,0,0,-2,-2,0,0,2"
280 NEXT X,Y
290 PRINT #-2,"A":END
```



Write: ADVENTURE

***Peter Gerrard** in a world of describable adventure!*

NOW that our adventure program has reached the stage where it can interpret sentences typed in by the player, or at least break those sentences down into a succession of words and numbers that the program itself can use, we reach the point where we have to supply a vocabulary. In other words, we need to give the program a list of all the words that it is going to use and understand. By that I don't mean words like 'the', 'and', 'it', or whatever, but those words that will enable a player to solve the game.

Such words as 'look', 'examine', 'open', and so on, should be common to every adventure, and in time to come we'll be programming those into our adventure's vocabulary. For now we need to concentrate on the ones that serve to distinguish one game from another, those that are particular to each individual program, and in order to do that we need to look very closely at our location descriptions.

Later on we'll be looking at ways of compressing the text so that we can get in as much detail as possible, but since the long description have to be typed in at some point anyway, we'll start from there.



Sample Descriptions

Take the following, which comes from an adventure of mine. All you need to know is that you've been exploring an old mine, and you are approximately half way down a decaying mine shaft: the outside world is still visible, since you are not yet that far underground. It goes something like this:

Up above you can make out the exit from the shaft, with a couple of stars twinkling as clouds scud by acrosss your line of vision. Further down, the shaft disappears into an inky blackness. Any stones that you dislodge fall for quite some time before hitting the bottom with a quiet crash. If you fell, you would probably make a slightly louder noise.

This is what we might term an 'interim' description, since there is nothing here for the player to do other than to go on or retreat. Nevertheless, these descriptions are important, and help considerably in adding to the flavour of the game. It isn't every location that can have a problem to solve, and no game should really be like that: you need to give the player some sense of progress as he overcomes a particularly difficult section. After spending day upon day struggling over something, it is very disheartening to move one location further on and be brought to a halt yet again!

In the above description the player is told everything that he needs to know, but it does go slightly beyond the 'you are in an old mine shaft' type of thing that featured (and sadly still does feature) in many a game. In the latter there is no sense of atmosphere whatsoever, you are simply solving a set of problems rather than exploring a brave new world. However, by using just a few words more than normal (and if we're going to compress things this doesn't make life any more difficult) we can give the player the true feeling of being there.

The line about '*If you fell, you would probably make a slightly louder noise*' tells the player all he needs to know about the depth of the mineshaft (albeit in a humurous kind of way) without going off into reams and reams of 'purple prose', as people are fond of calling that sort of thing. Little details like the stars overhead, the clouds scudding by, all add to the description and the atmosphere produced by it. The player can really begin to imagine that he's there, rather that sitting in front of a computer solving logic problems.

Humour

Personally I think that humour is essential to an adventure game, provided that it isn't overdone. Of course, some games rely almost totally on humour and a penchant for terrible puns ('*You are looking at a field of lambs. You feel sorry for them, knowing that in a couple of months time they're going to be taken away and slaughtered. They have gambolled and lost' sort of descriptions*), although I can't pretend to be overkeen on games like that (despite, as anyone will tell you, a liking for the pun!). But a little dab of humour here and there never goes amiss.

And, like certain comedians, it is possible to be humourous without resorting to foul language, which some adventures regrettably do. There is no excuse for that sort of thing. By all means have 'swear word' detector built into your parser, because if people like to type that sort of thing then they will do, and you can't print out anything that is coarse or crude. You never know who might be playing the game.

So with all that in mind, and sticking to uncompressed room descriptions for now, let us get our player moving around in his adventure world.



Player movement

First of all, you need the room descriptions. For now, if you haven't got 64 such descriptions lying handily around, or you can't be bothered typing them all in, we can use a simple *'YOU ARE IN ROOM NUMBER'* message, which at least will tell us that we are managing to move the player from one place to another, provided of course that we follow the message with the room number itself.

However, the way in which we could do that (44 PRINT 'YOU ARE IN ROOM NUMBER'CP for example, where CP gives us the player's current position) is markedly different for the player to read (44 ON CP GOTO ..... etc, where the dots would be replaced by line numbers indicating which line contains which room description, is just one way of doing it), so do leave a nice wodge of code free so that we can later put the room description in there.

But the problem is, how do we get a value for CP? Like this, dear reader.

Let us assume that the player is going to start the game at location number one. At the start of our program, then, we would have a line declaring CP to be equal to 1. Then, we could have a group of lines like the following (this is also assuming that you're using the parser from an earlier column, although that's up to you of course!):

```
6 VB$(5)="N":VB$(6)="S":VB$(7)=
  'E":VB$(8)="W"
1001 IF VB>5 AND VB<10 THEN 1050
1050 IF (VB=6 OR VB=8) AND CP<64
  THEN CP=CP+1:GOTO 44
1051 IF (VB=5 OR VB=7) AND CP>0
  THEN CP=CP-1:GOTO 44
1052 PRINT "YOU CAN'T GO THAT
  WAY.":GOTO 44
```

These are just being used as examples, and don't have to be followed slavishly.

From the above, you can see that we've defined four verbs to be equal to the four compass points N, S, E and W. You might like to include VB$(1)="NORTH", VB$(2)="SOUTH", and so on, to give the player the option of typing either 'n' or 'north' when he wants to move in that direction. Then, line 1001 checks to see if the verb typed by the player corresponds to one of these movement verbs, and if it does then go to line 1050. Change the VB>6 to a VB>0 if you've added the longer versions of the words.

Now, lines 1050 to 1052 are most definitely NOT the sort of thing you'd want to include in your adventure! But, as a starting point, it will let you check all your descriptions. Here, if the player has typed in verb 6 or verb 8 (ie he wants to go south or west) and the current location number is less than 64 (this assumes a 64 location adventure) then increases it by one and go back to line 44 where we could print up either our YOU ARE IN ROOM NUMBER CP message, or go off to the line that prints up the CPth room description. Similarly, if the player has typed in verb 7 or 9 (ie he wants to go north or east) and the current location number is greater than zero then decrease it by one and go back to line 44 again. Finally, if none of these conditions are true then tell the player that he can't move anywhere.

This, as I said, is only useful in allowing you to check your room descriptions, since we don't want players moving happily throughout the entire game without a care in the world or a problem to solve. So, we must have some way of controlling movement and getting the player going in the right direction.

If we define an array P%(64,4), we can achieve this control. Suppose that from location 1 the player could go north to nowhere, east to location 2, west to location 3 and south to location 4. Highly improbable, but it will do as an example. We could then define P%(64,1) to equal zero, P%(64,2) to equal 4, P%(64,3) to equal 2 and P%(64,4) to equal 3. In other words, we use the four elements of the array to represent the movement directions north, south, east and west: the order in which the verbs were originally typed in, if you recall. Thus P%(CP,MO) would give us the location that a player would end up in if he were to move in the MO direction from location CP.

It's easier to see by example:

```
1050 VB=VB-5:IF P%(CP,VB)<>0 THEN
  CP=P%(CP,VB):GOTO 44
1051 PRINT "YOU CAN'T GO THAT
  WAY.":GOTO 44
```

The value of VB lies in the range 6 to 9, although if you were also using the values 1 to 4 you'd have to do another check. All we want to end up with is a number in the range 1 to 4, representing north, south, east and west respectively. Then, if P%(CP,VB) is not equal to zero, that's where the player is going to end up. So we put him there by declaring the new value of CP to be equal to P%(CP,VB) and go to line 44 to inform the player of where he's got to.

If the value of P%(CP,VB) IS equal to zero then there is no movement allowed in that direction (over a cliff, or into a bottomless chasm, perhaps) so again we tell the player so and go off to line 44 again.

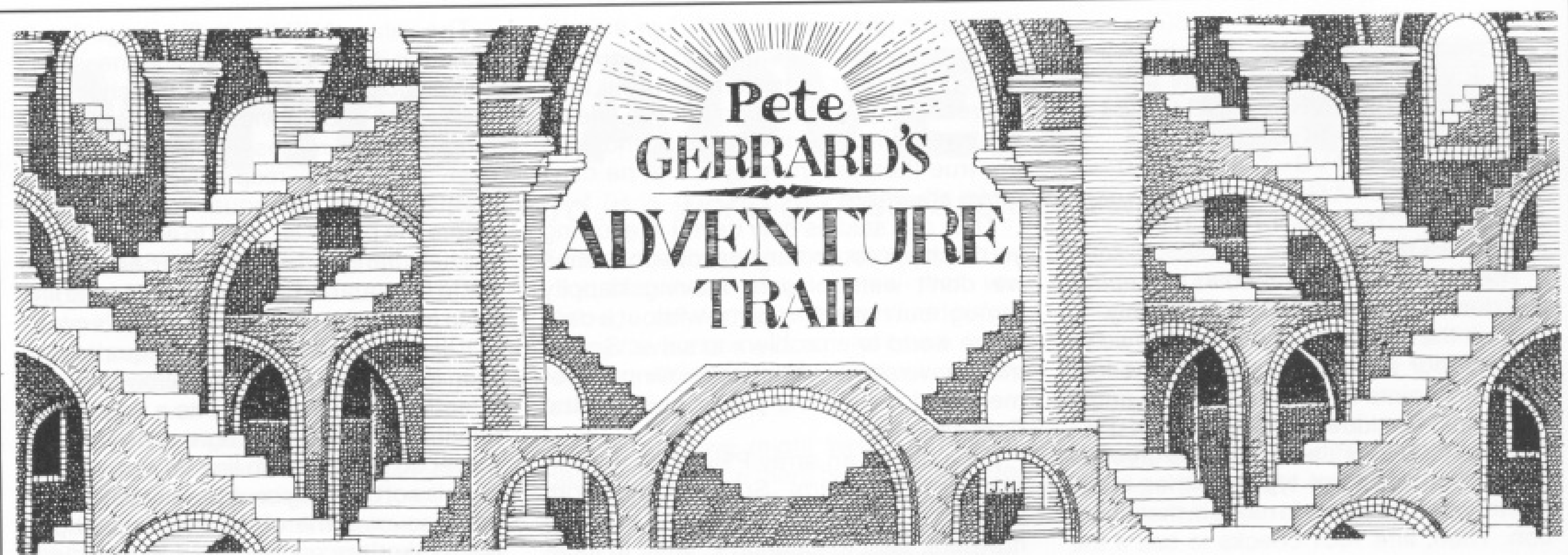
This sort of thing could easily be extended to cover some of the more usual problems that are encountered in an adventure. For example, the need to carry a torch in certain locations, say locations 40 to 51. If the torch was being carried we could have a flag TC set, and by altering the lines a little we could have something like:

```
1050 VB=VB-5:IF (CP>39 AND CP<52)
AND TC=0 THEN PRINT "ONE MOVE
AND YOU'LL FALL INTO A PIT":GOTO 44
```

and have the rest of the routine do the check for movement allowed or not allowed as the case may be.

Conclusion

By this straightforward use of an array we can effectively control the movement of the player through our adventure, as well as setting up checks for torches or anything else you might care to include.



I'D like to start off this month's *Adventure Trail* with a promise and an apology. First of all, I promise not to mention any sporting event ever again (unless it involves Wigan and they've won!). The time lag between my writing this and your reading it is such that anything I mention is lost in the mists of antiquity by the time it reaches you, and I've received a mild reprimand from a person who shall remain nameless but knows who he is because he asked my brother Mike to pass the message on (hello there!) to the effect that he didn't have the faintest idea what I was on about. Many people say this, but the fewer the better and so no more sports.

Secondly, apologies to anyone who hasn't yet received the solution to an adventure. I'll be with you as sooon as I can (ie when I've finished this column). Sorry, folks.

But enough of this taradiddle, pass those waffles and let's get on to the trail of the Dragon.

Sad News

We Gerrards seem to have the power of life and death over magazines. No sooner do we mention one and it disappears forever. *Insight* bites the dust, so bad luck to Ron Dawson, and after a fleeting appearance in last month's column so has *Adventure Contact*. This is even worst than usual because it was produced in Wigan so commiserations to Pat Winstanley. However, its sister magazine, *Adventure Probe*, is alive and well for the next three and a half years at least (or so I'm told). Sandra Sharkey, the editor, when not beating me at pool would love to hear from you at 78 Merton Road, Wigan, Lancashire WN3 6AT.

Better News

A letter from Joe Brincat is always good news, since this man has solved more Dragon adventures than have been written. With Joe being from Malta we get an international flavour to the column (and more to come, with Gibraltar lurking in the wings), so let's help my stamp collection and keep 'em coming.

In July issue of *Dragon User* the Expert printed Joe's map of *Total Eclipse*, but neglected to include the key to the map. So, ever one to make amends, you'll find it attached somewhere to this column. (*Hhhmmph. No you won't. I don't know if you're going blind or I'm going mad, but there's a perfectly good key in the top left hand corner of mine. Granted it's a bit faint. If we get six more letters, and we can find the original, we'll run the key again sometime. — Ed.*)

Now completing this mammoth game is an incredible feat of adventuring, but alas there be still problems. There is, for example, the abandoned craft, which nobody seems to have found. Salvage and Invest won't work, so on behalf of adventures everywhere, HELP! the producers of the game, Fenmar, are as elusive as the abandoned craft, so if anyone can help me and Joe please write in. Joe's address is 73 Annunciation Street, Hamrun, Malta 28-06-87, and he'll be happy to help anyone with *Total Eclipse* (apart from the above problems, of course). Also, at the completion of Universe 1 the following words appear on the screen:

"Well done commander you have completed Universe 01. Place Universe tape 2 in player — prepare tape player — prepare tape player — press Enter when ready".

Where is Universe 2, this is the question. To quote Joe's letter "I will pay its value to whoever sends it (to me) first and when one day I've finished mapping it, he will be the first one to get the map". Okay you lot, now there's an offer you can't refuse. Incidentally, Joe's version of *Total Eclipse* was 1.3, so perhaps any later ones have a more easily found abandoned craft. Hopefully, we shall see.

Staying in this universe for a while, Donald Morrison is having problems with *Syzygy*, saying that when he enters the co-ordinates for the planet he gets transported into space. Well, are you entering the correct co-ordinates? These are 0-4-1-5, so enter that number then PRESS LEVER and he ho and away you go.

Going onto parallel universes we arrive at Hemel Hempstead and Graham Burton, who imparts some words of wisdom concerning Operation Sapras. And I quote "To find antidote go south, east, east, east, then west in the 'green and brown'; Ron Tripp is of further use when you find him: book stalls can be useful; take note of what the man in Aberdeen says". It must mean something! Of course it does.

Alas, Graham is not on a par with the Pope, and does make the odd mistake. Like erasing two of the machine code routines on *The Ring of Darkness*, thus making it impossible to solve. Ah well. Problems with *Vortex Factor* as well, so let me tell you what the hair pin is not for picking the lock of the north door, you must enter the hole, NOT go north, and then use the hair pin. As far as I can tell the aquarium is not essential to completing the game, so if I were you I'd make for that hole and continue from there: you'll find at least two more treasures by looking at the chest that you'll find there.

A plea for help from Neil Davies, of 4 Grinshill Drive, Crowmoor, Shrewsbury, Shropshire, SY2 5JE. He's completed *Trekboer*, *Sea Quest*, *Calixto Island*, *Jerusalem Adventure*, *Ultimate Adventure*, *Shenanigans* and *Syzygy*, and will help anyone who sends him an SAE. However, he also wants to know if there's anyone in the Shrewsbury area willing to swap games. Okay Shrewsbury folk, get writing.

All right, all right, Gibraltar here we come. Andrew Powell, 1 Old Naval Hospital, Gibraltar, to be precise, sends a letter that's a curious mix of grovel and thinly veiled threat. Grovel in the form of worship (this is the stuff) and threat in the message that if I don't help him out he'll send the boys round. Why does no-one wever send the girls round, that's what I want to know (cue several thousand letters of complaint!). Anyway, the lad is stuck on *Return of the Ring*. He's not the only one, so if anyone has a solution, Andrew and I would be pleased to see it. So would Graham Burke and a few others as well.

Having cunningly worked Graham Burke into the conversation, he is having a problem or three with *Juxtaposition*, and reasonably near the start as well, Pity on the lad, so making sure you've got everthing before you get there (like blue key card, sleeping pills and anything else you might find lying around after short-circuiting the droid) go to the monorail platform (and don't go into endless rez zones), wait for the car and board it when the doors open. Get the camera from the photo gallery (only take food from the dinner droid when you need to eat). Go to the ante-room and put sleeping pills in the caretaker's coffee. Get the white key card and cup from caretaker and unlock the door. Get the breather mask, sticky tape and soldering iron, go back to the street

area and sort the rest out for yourself! That's more than enough to get you well into the game, I think.

Moving rapidly onto Ipswich, where Nick Webb is a man lost without matches in *Black Sanctum*. Well, me old son, try examining the mantlepiece in the cabin, and lo and behold, got a light, boy? No, but I've got a heavy woman.

Despite everything, despite hints and tips in various columns, I am getting just as many letters about this game as ever, whihc means either that people are new to the column, or they don't read it, or they've just acquired the game. In order to stop droves of you sending in requests for hint sheets, my postman suffering heart failure and my printer ribbon being worn to a ... well, whatever it is that printer ribbons get worn to, here is the COMPLETE SOLUTION to *Aquanaut 471*. Don't read it if you don't want to know. It's presented in the order in which you need to do things, and comes courtesy of Michael Edwards of Broomsoft. Here goes...

1) From the very start, go D, E, N, E, S, E, N, N, D, E, D, D.

2) Play the (expletive deleted) bubbles game.

3) Find the memory GRID and the MIRROR.

4) GO POSTER in the storage room (the mirror protects you against the lasers).

5) FIX COMPUTER WITH GRID then RUN COMPUTER.

6) Take plastic CARD. Go W, N, E. Take the BLASTER. Go W, S, E. Tell Huey-12 to FOLLOW ME. Go S, E. Drop the MIRROR.

7) Go to the force field with Huey (if he stops following you find him and tell him to follow you again and tell him to GO NORTH through it. Go N, W, S. Take the FLASHLIGHT. Tell Huey to follow you.

8) Go to the tool chest. Open it and look at it twice. Take both items that you find there.

9) Go to the sliding door. OPEN DOOR WITH CARD then GO DOOR. OPEN BOX WITH CROWBAR. Look twice, drop the crowbar and take the AIR HELMET.

10 Go to the (expletive deleted) robot game. Go N, D. LIGHT the FLASHLIGHT then N, D (past more robots), OPEN DOOR WITH CARD then U, S, U. UNLIGHT FLASHLIGHT then DROP CARD.

11) Find Huey and go to the hatch. Open hatch, tell Huey to follow you (he MUST keep following you) then GO HATCH.

12) Find the mutant. SHOOT him then drop the BLASTER. Go W. Find the box cavern and take the seaweed with the plant cutters. Drop the cutters then find the caves, and remember Huey must be following you.

13) Go in the caves. Light the flashlight and go S, E, E, N, W. Take the MUSHROOM and go U. Unlight the flashlight. Find the module and tell huey to LIFT MODULE.

14) Take the module. Combine the SEAWEED with the MUSHROOM to make a radiation pill.

15) Return (you don't need Huey now) to the base, remembering to UNLIGHT FLASHLIGHT when you don't need it.

16) Go past the security robots, N, D,LIGHT FLASHLIGHT, N, D. EAT the PILL. GO DOOR then FIX REACTOR WITH MODULE. Return to the poster room, get mirror, go to the poster, N, PRESS BUTTON and ..... END!!!!!!!

Huey-14's message is:

To save the director you must:
1) Make a radiation vaccine.
2) Fix the reactor.
3) Press the button.
Signed ... Huey-14

Right, me again. No more letters about *Aquanaut 471*, okay chaps and chapesses?! See you next month.

CLASSIFIED ADS

DRAGON 32, perfect condition, unused, joysticks, books, DU back issues, 30 games and other software. £90 ono. T. Eccles, 23 Holly Hill Lane, Sarisbury Green, Hants. Tel: Locks Heath (04895) 2096.

PYRADVENTURE–AMENOPHIS III. Enter the Lost Tomb of King Tutankhamen's Father and travel back in time 3000 years in this thrilling adventure game. Please send £2.95 plus 50p P&P to: 23 Ashby Road, Thurton, Norwich NR14 6AX.

DRAGON 32, with joystick, cassete recorder, leads and manual. Also software including Word Processor, Database, Forth and many games, £50. Tel: Portsmouth 478920.

DRAGON GAMES at cheap prices. All original. Send S.A.E. for list to Richard Heath, 66 Newcastle Road, Leek, Staffs. ST13 5RU. Tel: Leek 383175.

DRAGON 64 DragonDos, Dragon twin disc-drive — all in working order, may split. Total £150. Tel: 061 665 1727.

DRAGON 32, Analogue joysticks, reams of software (nearly 200 titles), light pen. All in good condition, will split if necessary. All worth about £1000, going for £500 ono. Tel: Gt. Missenden (02406) 2535.

WANTED Working disc-drive and controller in the Nottingham/Derby area. Tel: (0623) 792226.

PROLOG interpreter for OS9. Full-feature 5th-generation language, with 100-page manual and utility module. £29.95 from METASOFT, 4 Pinehurst Walk, Orpington, Kent BR6 8DD.

LARKSPUR Graphic Adventure Trilogy, Parts 1 and 2, available now. £2. Tel: (0772) 774485.

DRAGON 32. + joysticks, books £45 ono. Tel: 061 861 8456 (evenings).

SUPERDOS disc drive controller kit (excluding case) £55; tested, working Dragon 32 (no lid) £38; untested, complete, could be working! £35; Most Dragon spares inc. 74LS783 – £10.25, 6809E, 41464, transformers, CPUs, PSU'PCB. Phone for details – Nick Spiers. Tel: (0892) 44070 (evenings).

DRAGON 32 for sale. In original conditions with box and manual. Tel: (0935) 75319.

LOW COST Dragon Disc Systems (Drive and Interface) from £130. For a 3½", 250Kb system (plus VAT and postage). Also 5¼" and 40/80 Track Drives, and Printers. Send a *long* S.A.E. to: DATASPAN LTD., 10 Pine Drive, Hawley, Camberley, Surrey. Tel: (0276) 31507.

HERE'S MY CLASSIFIED AD.
(please write your copy in capitals on the lines below)

Name . . . .

Address . . . .

. . . . **Tel:** . . . .

CLASSIFIED ADVERTISING RATE: LINE BY LINE 35p per word.

Please cut out and send this form to: Classified Department, Dragon User, 12–13 Little Newport St, London WC2H 7PP.

Odd One Out

*When **Gordon Lee** gets going, he goes one and one and one*

QUESTION: Which of the following is the odd man out?

7, 15, 23, 29, 32, 44, 50, 67, and 83.

Answer: No. 44, 'cos its the only one that includes fried rice!!

End of mathematical joke and on to some larger numbers!

Consider that series of numbers which are made up of the same digit repeated a specified number of times. These are known as 'monodigit' numbers, and particular, much interest has been shown in repetitions of the digit '1' — the term 'repunit' being used to describe numbers of this type, and it will be these that we will be examining here.

For convenience, we can use a type of algebraic shorthand to refer to these numbers, using the form Ry, where y denotes the number of ones present. So, for example, R_2 will denote 11, and R_{17} indicates 11111111111111111.

The main interest in these numbers has been in determining their factors, and in particular discovering which of the series is prime. R_2 is obviously prime, but the next prime number in the series is not reached until R_{19}. The intervening numbers are all composite, but establishing their factors is not at all easy. For example, even the modest R_{17} already mentioned above, has only two factors: 2071723 and 5363222357 — and if you don't believe it, just multiply them and see! The only other reasonable low repunit which is known to be prime is R_{23}, after which R_{317} is the only prime for values less than R_{1000}. Enigmatically, there are large gaps in the knowledge about these numbers. Most of the series have been proved to be composite although in many cases the factors themselves have not been established.

An associated problem involving repunits is as follows:

Take any prime number p, and find a value v, such that the product of p times v is comprised only of ones. As a simple example, the prime 3 when multiplied by 37 will result in 111. Clearly, this is impossible for the two primes 2 and 5 (can you see why), but of the rest such a value exists for v, although this number may be exceedingly large!

The table shows the factors required for all primes (except 2 and 5) under 100. The number in column y indicates the number of ones in the final product — ie the value of y in the formula Ry. Of particular interest are those primes marked with an asterisk. With these, the value of y is one less than the prime itself. Mathematically, for this to happen, 10 has to be primitive root of the prime in question. These numbers turns up in a different guise if we were to evaluate the reciprocals of primes. Except for 2 and 5, all primes have reciprocals with digits which repeat in cycles.

For example, 1/7 = 0.142857 ..... and so on to infinity. This cycle of six digits is reflected in the y column of the table, where, except for the prime 3, the column indicates the number of digits in each cycle:

1/11 = 0.09 09 09 09 09 ....
1/13 = 0.076923 076923 ....
1/17 = 0.0588235294117647 ....

and so on.

The competition question this month is concerned with the prime 34543. What we want to know is the smallest number by which we can multiply 34543 in order to obtain a product consisting entirely of ones. Now, as 10 is a primitive root of 34543 this product will contain 34542 ones, and consequently the number that we are after will be almost as large. So all that we need is the ten digit sequence which occurs at position twenty-thousand and one to twenty-thousand and ten.

Remember, the number itself has 34537 digits — commencing with 3216602817 and wnding with 9772779177.

3216602817...............(??????????)..........
....9772779177

Can you supply the missing digits?

Prime	y	Complementary Factor:
3	3	37
* 7	6	15873
11	2	1
13	6	8547
* 17	16	65359477124183
* 19	18	5847953216374269
* 23	22	4830917874396135265 7
* 29	28	3831417624521072796934865 9
31	15	3584229390681
37	3	3
41	5	271
43	21	258397932816537467 7
* 47	46	2364066193853427895981087470449172576832151 3
53	13	20964360587
* 59	58	1883239171374764595103578154425612052730696798493 4- -086629
* 61	60	1821493624772313296903460837887067395264116575591 9- -85428051
67	33	1658374792703150912106135986733
71	35	15649452269170579029733959311424 1
73	8	152207
79	13	14064697609
83	41	1338688085676037483266398929049531459 17
89	44	124843945068664169787765293383270911360799
*' 97	96	1145475372279495990836197021764032073310423825887 7- -4341351660939289805269186712485681557846506 3

Prize

DON'T be stingy, we said to Microvision Software, and so they gave us *Miser's Dream*. Is there a hint in there somewhere? Whether or no, there are 20 copies of this program for the twenty best entries to this month's competition.

Rules

Pull out your digits and solve the puzzle. Then (and not before) send us your solution, with any notes you want to include, with a printout of your program (no tapes, please) and your name and address, in an envelope marked SEPTEMBER COMPETITION.

To make everything fairer (no, you can't leave the solution out), please complete the tiebreaker "As soon as I've made my first million ..." and sent it to us with your solution.

June winners

Lots of entries. We didn't reject people who used the 'leading zero' as long as they had the other answers right. Gordon may have something to say about that eventually — some of the entrants certainly did. The winners are:

John Machin of Troon, Keith David of Crawley, Richard Long of Camberley, Jake Anderson of Edinburgh, Denis O'Mulloy of Comberton, Phil Callaghan of Stafford, R. Raine of Sapcote, Paul Weedon of Wotton-under-Edge, Dave Lardner of Rutherglen, Fred Taylor of Middlesbrough, Austan Henderson of Bromsgrove, Graham Barber of Sutton Coldfield, S.P. Greenard of Ingatestone, Terry Potter of Chiseldon, Phil Davies of Longhoughton, Andrew Powell of Cambridge, John Smallwood of Preston, Alan Thomas of Staplehurst, Clive G. Scott of Ashstead and D.J. Gray of Middlesbrough.

And the mystery prize, kindly donated at short notice by small but devoted software house **Broomsoft**, is a set of free vouchers and a set of discount vouchers. Thank you, Andrew. The free vouchers will go to the worthiest programs, and the rest to the runners up. We aren't going to announce who is who here — all the winners sent in a high standard of entry.

And some very silly (and even poetic tiebreakers) which gave us lots of laughs. Favourite from Clive G.Scott: "Them wuz the days. Why, when I was seventeen, computer programs had to be saved on wax cylinders." But there were lots of other good ones.

Solution

This month's solution appears on page 27.

The Answer

This is Gordon Lee's own solution to the June competition **see page 26 for results**

ANSWER: There are two numbers which will produce the longest sequence possible (of eight steps). These are 909 or 999.

9095 (= 535 * 17)	9996 (= 588 * 17)
0952 (= 56 * 17)	9962 (= 586 * 17)
9520 (= 560 * 17)	9622 (= 566 * 17)
5202 (= 306 * 17)	6222 (= 366 * 17)
2023 (= 119 * 17)	2227 (= 131 * 17)
0238 (= 14 * 17)	2278 (= 134 * 17)
2380 (= 140 * 17)	2788 (= 164 * 17)
3808 (= 224 * 17)	7888 (= 464 * 17)
808?	888?

Solution: The program begins with all starting values in the range of 100 to 999. These are held in the FOR/EXIT loop at line 110. For each test, this value is transferred to variable 'X', and it is on this variable that the operations are performed. Loop 'D' at line 140 appends each digit in turn, and the result (in variable 'Z') is tested for divisibility by 17. If an exact multiple is found, the value currently in 'D' is transferred to variable 'Q'. This variable (Q) is set initially to the value -1 (line 130) and if it remains at -1 when tested at line 180, then no multiple has been found for the current sequence under test. However, if a multiple is found the new number is generated at line 190, and the right-hand three digits are extracted at line 200 by means of a string variable. This is then converted back to numeric variable X, and the whole process is repeated from line 130. A 'count' is kept of the number of steps possible for each of the starting values, and this number is checked against the current maximum at line 250.

If the program is run, it computes this maximum step as being 8. It is then necessary to re-run the program a second time, having amended line 250 to read:

```
250 IF T=8 THEN PRINT S
```

Line 250 should also be deleted. This second program will then print those starting values for which a step value of 8 has been achieved.

```
100 MAX=0
110 FOR S=100 TO 999
120 X=S:T=0
130 Q=-1
140 FOR D=0 TO 9
150 Z=X*10+D
160 IF Z/17=INT(Z/17)
    THEN Q=D:T=T+1
170 NEXT D
180 IF Q=-1 THEN 250
190 X=X*10+Q
200 X$=STR$(X):
    X$=RIGHT$(X$,3)
210 X=VAL(X$):GOTO 130
220 NEXT S
230 PRINT"MAXIMUM
    STEP=";MAX
240 END
250 IF T>MAX THEN
    MAX=T
260 GOTO 220
```

Communication

Problem: Where can I obtain a program in Basic to rotate images in X-Y-Z planes, the points being obtained from an array in Pts. X,Y and Z in high resolution? Help.
Name: Dave Bonnett
Address: Raiffeisenstr. 8A, D4450 Lingen/Biene, Germany.

Problem: I need to connect my TRS-80 DMP 120 Printer to my Dragon. Is there a suitable interface?
Name: Chris Vollbracht
Address: 110 Alpine Rise, Styvechale Grange Estate, Coventry, W. Midlands CV3 6NR.

Problem: Does anyone have the cassette version of *Empire* by Shards Software which they are willing to sell?
Name: Karen Carpenter
Address: 41, Glenleary Road, Coleraine, Co. Londonderry, N. Ireland.

Problem: Desperate. Frantic. Pulling out of hair. Near to nervous breakdown type plea. To borrow, buy or even a teensy weensy peek at a copy of *The Advanced Programming Guide for DeltaDOS* book. Very ditto the manual which accompanies Morrison's Pascal Compiler. Also 747 Flight Simulator manual. Original publishers have none left, and mine have been nicked. I have photocopying facility and will pay all postage and packing.

Exceeding desperate for a dead or dying DragonDOS cartridge.
Name: Paul J. Read
Address: 7, Fairview Avenue, Whetstone, Leicester LE8 3JQ.

Problem: Does anyone have a Dragon's Claw interface (and possibly a Snap camera and software) that they are willing to sell or lend?
Name: Andrew Bell
Address: 7 St. Pauls Road, Cambridge.

Adventure Contact

To help puzzled adventurers further, we are instituting an Adventure Helpline — simply fill in the coupon below, stating the name of the adventure, your problem and your name and address, and send it to Dragon User Adventure Helpine, 12/13 Little Newport Street, London WC2H 7PP. As soon as enough entries have arrived, we will start printing them in the magazine.

Don't worry — you'll still have Adventure Trial to write to as well!

Adventure ....
Problem ....
....
Name ....
Address ....
....

Adventure Contact

Adventure: *The Ket Trilogy*
Problem: How do you get past the zombie?
Name: Paul Dean
Address: 26, Millhouse Drive, Wymans Brook, Cheltenham, Glos. GL50 4RG.

Adventure: *Ring of Darkness*
Problem: How to board the cart, also how to get the keg from the jester.
Name: Aiden Devine
Address: 60, Nether Street, Finchley, London N12 7NG.

Adventure: *Syzygy*
Problem: I have found the transporter and co-ordinates. How to use the transporter?
Name: Kieron Williams
Address: 33 Mayflower Drive, Margord, Nr. Wrexham, Clwyd LL12 8LD.

Adventure: a) *Shenanigans* b) *Cricklewood Incident*
Problem: a) How do you safely cross the rainbow? b) Which command allows you to board the bus?

Name: Chris Vollbracht

Address: 110 Alpine Rise, Styvechale Grange Estate, Coventry, W. Midlands CV3 6NR.

TURBOCHARGE YOUR DRAGON:
With our great value hardware and software:

BASIC 42
Extended BASIC for the Dragon 64

For Dragondos (please state version) £14.95

Run your Dragon in 64K mode, while retaining BASIC and DOS. Print on hi-res screen, using standard PRINT commands, and a screen layout of 24 rows of 42 columns. Other features include:

Alternative, redefinable character sets, control key for special characters, repeating keys, and commands in lower case, windows, CATCH command for automatic return to window, inverted video (green on black/black on green), true underlining and extra PRINT commands and functions.

LIBRARY lists commands and functions. Automatic startup of BASIC program. TEXT command for software compatibility. Still 23335 bytes free to BASIC. Patches for Dragondos 1.0. Can load in extra UTILITIES from disk:

HELP UTILITY £5.00
Extensions to BASIC 42 include change cursor character, scroll disable, pause listing, BREAK disable, improved TRON (allows single stepping), Detailed help and error messages.

SPOOL UTILITY £5.00
Use computer while printing. 3.5K print buffer TYPIST program turns Dragon into typewriter.

ICONS UTILITY £5.00
Put icons in your program! Controlled by cursor or "mouse". Commands to define, clear, load and save icon positions and windows.

STRUCTUR UTILITY £5.00
Another first! Structured BASIC on the Dragon! Allows named procedures, improved loop controls by WHILE ... WEND, and REPEAT ... UNTIL etc.

DOS UTILITY £5.00
Make friends with your DOS! Enter all the main DOS commands, plus LIST, EDIT etc, and select files by cursor or "mouse".

WHAT THEY SAY ABOUT BASIC 42:

"probably the best step so far"
Dragon User

"an invaluable utility"
Dragon Update

HARDWARE

Memory Upgrades	**FROM**	**£35.00**
Superdos Cartridge		**£75.00**
40 Track Drive inc. Cartridge		**£180.00**
Superdos controller (chip only)		**£10.00**
Peaksoft joystick (ideal "mouse")		**£5.50**

DISK SOFTWARE FOR DRAGON 32/64/128 WITH DRAGONDOS/CUMANA DOS 2.0

• • NEW • •

Pixie (Mindsoft) £14.95
Icon-driven drawing program. Requires joystick.

• • •

DSKDREAM (Grosvenor) £19.95
The standard Dragon Editor/Assembler

D.R.S (Grosvenor) £9.95
Machine code database program

SOURCEMAKER (Pamcomms) £8.50
Disassembler for use with DSKDREAM

DISK-KIT (Pamcomms) £9.95
Sort out your disk problems

MACGOWAN SOFTWARE

PRINTER CONTROL* FROM £19.95
A text AND graphics processor

DUMPER* FROM £5.45
Relocatable screen dump program

COLOR PRINT* FROM £6.50
PMode 3 screen dump program

• • NEW • •

LIGHTPEN S/W* CASS £7.00 DISK £8.00
Upgrade includes screen dump

MONITOR/ASSEMBLER* CASS £12.00 DISK £15.00
Printer orientated

*Prices vary according to printer: please specify.

MONEYBOX (Harris) £14.99
Home and small business accounts

MAILBOX (Harris) £16.99
Selective mailing list program

SHAREBOX (Harris) £16.99
Manage your stocks and shares'

SALESBOX (Harris) £19.99
Balance B/F Sales Ledger'

BILLSBOX (Harris) £19.99
Balance B/F Purchase Ledger

CASHBOX (Harris) £19.99
Double-entry Nominal Ledger

STOCKBOX (Harris) £19.99
Full-featured Stock Control

ORDERBOX (Harris) £16.99
Invoicing linked to Sales or Stock

Cheques/P.O.'s/Further details/dealer enquiries to:

HARRIS MICRO SOFTWARE
49 Alexandra Road, Hounslow, Middlesex, TW3 4HP Tel: (01) 570 8335

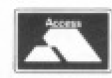
DR57

S.P. ELECTRONICS

Complete Dragon Spares and Repairs Service

Citizen 120D Dot Matrix Printer (NLQ)	£199.00
Ferguson Green Screen Monitor	£64.00
Philips Green Screen Monitor (7502)	£85.00
Philips Amber Monitor	£95.00
Parallel Printer Cable	£12.90

S.P. ELECTRONICS



48 Limby Road, Hucknall, Notts
(Nottingham 640377)

VISA

DR14

DON'T FORGET ! !

THE N.W. ENGLAND DRAGON SHOW/CONVENTION

AT:— ROCHDALE
ON:— SATURDAY 12TH SEPT '87
TIME:— 10.00 AM — 4.00 PM
VENUE:— BISHOP HENSHAW UPPER SCHOOL, SHAW RD., (OFF MAIN ROCHDALE — OLDHAM ROAD), THORNHAM, ROCHDALE.

(ADULTS £1.50, CHILD £1.00)

THERE WILL BE SOMETHING FOR EVERYONE.

MAJOR SOFTWARE/HARDWARE FIRMS, USER GROUPS, DEMOS, TALKS, FORUM, COMPUTER CLINIC, PRIZE DRAWS, SNACKS, LICENSED BAR APPLIED FOR.

EASY ACCESS FROM M62 (JCT. 20)

ENQUIRIES TO:—
PULSER SOFTWARE
(0706-849189)
(BRIAN O'CONNOR)

DR58